



BACHELORARBEIT

Frau
Jana Weigel

**Konzeption und Realisierung
eines Komplexpraktikums zum
Thema Auszeichnungssprachen**

2012

BACHELORARBEIT

Konzeption und Realisierung eines Komplexpraktikums zum Thema Auszeichnungssprachen

Autorin:
Frau Jana Weigel

Studiengang:
Medientechnik

Seminargruppe:
MT09wD-B

Erstprüfer:
Prof. Dr.-Ing. Frank Zimmer

Zweitprüfer:
Dipl.-Ing. Sieglinde Klimant

Einreichung:
Mittweida, 26.07.2012

BACHELOR THESIS

Design and implementation of a complex placement on markup-languages

author:

Ms. Jana Weigel

course of studies:

media engineering

seminar group:

MT09wD-B

first examiner:

Prof. Dr.-Ing. Frank Zimmer

second examiner:

Dipl.-Ing. Sieglinde Klimant

submission:

Mittweida, 26.07.2012

Bibliografische Angaben

Weigel, Jana:

Konzeption und Realisierung eines Komplexpraktikums zum Thema Auszeichnungssprachen

50 Seiten, Hochschule Mittweida, University of Applied Sciences,
Fakultät Medien, Bachelorarbeit, 2012

Abstract

Diese Bachelorarbeit beschäftigt sich mit der Konzeption und Realisierung eines Komplexpraktikums zum Thema Auszeichnungssprachen im Kontext des Themenumfeldes Energie und Umwelt. Ziel der Arbeit ist es, den Studenten fundamentale Kenntnisse im Umgang mit Auszeichnungssprachen zu vermitteln und sie zu befähigen, dieses Wissen anzuwenden. Dafür soll hier auch der Blick auch auf nachhaltige Lernmethoden gerichtet werden.

Inhaltsverzeichnis

Inhaltsverzeichnis	V
Abkürzungsverzeichnis	VII
Abbildungsverzeichnis	VIII
Tabellenverzeichnis	IX
Vorwort	X
1 Einleitung.....	11
2 Auszeichnungssprachen.....	12
3 Vorbetrachtungen	15
3.1 Die Zielgruppe.....	15
3.2 Methodische und didaktische Aspekte	16
3.3 Die Inhalte.....	18
3.4 Der Aufbau des Komplexpraktikums	18
4 Konzeption des Komplexpraktikums.....	20
4.1 Schwerpunkte	20
4.1.1 Komplex I - HTML	20
4.1.2 Komplex II - HTML und CSS	21
4.1.3 Komplex III – XML und CSS.....	21
4.1.4 Komplex IV – DTD, XML-Schema	21
4.1.5 Komplex V – XHTML, HTML5, Zusammenfassung	22
5 Umsetzung des Komplexpraktikums.....	23
5.1 Verwendete Technik	23
5.2 Komplex I.....	25
5.2.1 Die Vorbereitung	25
5.2.2 Das Praktikum.....	28
5.3 Komplex II	29
5.3.1 Die Vorbereitung	30
5.3.2 Das Praktikum.....	34
5.4 Komplex III.....	35
5.4.1 Die Vorbereitung	35
5.4.2 Das Praktikum.....	38
5.5 Komplex IV.....	41

5.5.1	Die Vorbereitung	41
5.5.2	Das Praktikum.....	42
5.6	Komplex V.....	43
5.6.1	Die Vorbereitung	43
5.6.2	Das Praktikum.....	45
5.7	Testphase	46
6	Fazit und Ausblick	49
	Literaturverzeichnis	LI
	Anlagen.....	LIV
	Eigenständigkeitserklärung	LXV

Abkürzungsverzeichnis

CSS	Cascading Style Sheets
DTD	Document Type Definition (zu Deutsch: Dokumenttyp - Definition)
HTML	Hypertext Markup Language (zu Deutsch: Hypertext-Auszeichnungssprache)
MIME	Multipurpose Internet Mail Extensions - ermöglicht es, zwischen Sender und Empfänger Informationen über den Typ der übermittelten Daten auszutauschen und gleichzeitig eine für den verwendeten Übertragungsweg sichere Zeichenkodierung festzulegen.
PHP	Hypertext Preprocessor - Script-Sprache, mit der sich dynamisch generierte Inhalte erzeugen lassen
SGML	Standard Generalized Markup Language - eine Meta-Sprache, die das Definieren von Auszeichnungssprachen mit Hilfe von DTDs erlaubt
URL	Uniform Resource Locator - Adressierungsform für Internetadressen, die vor allem innerhalb des World Wide Web (WWW) zur Anwendung kommt ¹
W3C	World Wide Web Consortium - internationale Organisation, welche zusammen mit der Öffentlichkeit daran arbeitet, Web-Standards zu entwickeln, wie etwa HTML oder XML
XHTML	Extensible Hypertext Markup Language (zu Deutsch: erweiterbare Hypertext-Auszeichnungssprache)
XML	Extensible Markup Language (zu Deutsch: erweiterbare Auszeichnungssprache)
XSL	Extensible Stylesheet Language (zu Deutsch: erweiterbare Formatsprache)

¹ DATACOM 2012, o.S:

Abbildungsverzeichnis

Abbildung 1 Ausschnitt der HTML Seite (1)	28
Abbildung 2 Übersicht über jährlichen Stromverbrauch der größten Stromverbraucher-Länder in Bezug auf das Internet	29
Abbildung 3 Rahmentypen mit CSS	33
Abbildung 4 Box-Modell	34
Abbildung 5 Übung zu XML-Regeln	36
Abbildung 6 Umsetzung der Tabelle in XML, Ansicht im Internet Explorer 8	40
Abbildung 7 Darstellung im Internet Explorer 8.....	40
Abbildung 8 nicht korrekte Darstellung im Mozilla Firefox 12.....	40
Abbildung 9 Screenshot der grafischen Erstellung des XML-Schemas in Eclipse	43
Abbildung 10 Video- Formate mit jeweiliger Browserunterstützung	45

Tabellenverzeichnis

Tabelle 1 Zeitumfang der Praktikumskomplexe	46
Tabelle 2 Zeitumfang der Praktikumskomplexe nach Anpassung.....	47

Vorwort

Das im Rahmen dieser Arbeit entstandene Komplexpraktikum für den Studiengang Energie- und Umweltmanagement ist im Anhang sowie auf der beiliegenden CD zu finden. Auf der CD befinden sich außerdem die umgesetzten Projekte samt aller Bilder, Videos und Codes sowie alle Lösungen für die Vorbereitungsfragen und –Übungen.

Um die Verwendung von Quellcode zu verdeutlichen, wird in der Arbeit hierfür eine andere Schriftart benutzt: `Codebeispiel`.

1 Einleitung

Ausgangspunkt dieser Arbeit ist die Neugründung des Studiengangs „Energie- und Umweltmanagement“ an der Hochschule Mittweida. Hieraus ergibt sich das Thema dieser Arbeit. Für den neuen Studiengang ist ein Komplexpraktikum zum Thema Auszeichnungssprachen zu konzipieren und umzusetzen, welches im Modul „Grundlagen der Informatik“ im ersten Semester stattfinden wird. Das Komplexpraktikum soll den Studenten fundamentale Kenntnisse im Umgang mit Auszeichnungssprachen im Kontext des Energie- und Umweltmanagements vermitteln und sie dazu befähigen, dieses Wissen zu festigen. Dabei wird ein Zeitfond von ca. 5 Veranstaltungen à 90 Minuten angenommen. Optional ist die Umsetzung als E-Learning-Angebot möglich und auch die Erweiterung bzw. Anpassung an Browser mobiler Endgeräte kann Bestandteil sein.

Um eine theoretische Ausgangsbasis zu schaffen, wird im ersten Kapitel der Arbeit in den Themenkomplex Auszeichnungssprachen eingeführt.

Der Hauptteil der Arbeit beginnt im dritten Kapitel mit einigen Vorbetrachtungen. Die Zielgruppe wird kurz analysiert, die Inhalte festgelegt und der genaue Aufbau des Komplexpraktikums erläutert. Anders als in anderen wissenschaftlichen Arbeiten, die sich mit ähnlichen Aufgabenstellungen beschäftigen, soll auch der Blick auf die Didaktik gerichtet werden.

Im vierten Kapitel folgt die Konzeption des Komplexpraktikums. Hier wird festgelegt, welche Inhalte genau behandelt werden und was die Ziele der einzelnen Komplexe sein werden.

Anschließend folgt im fünften Kapitel die Umsetzung des Komplexpraktikums, welche für jeden der fünf Komplexe dokumentiert und erläutert wird.

Nach einer Testphase erfolgt im sechsten Kapitel eine kurze Evaluierung, welche durch Tests im kleinen Kreis durchgeführt wird. Durch diese Evaluierung und ein Fazit wird die Arbeit abgeschlossen.

Im Anhang dieser Arbeit sowie auf der beiliegenden CD befindet sich das fertige Komplexpraktikum.

.

2 Auszeichnungssprachen

Eine Auszeichnungssprache (engl.: markup language) ist eine formale (nicht-natürliche) Sprache, die es ermöglicht, unterschiedliche Bestandteile eines Textes als solche zu kennzeichnen. Durch eine Auszeichnungssprache können beliebigen Textelementen auf deklarative Weise Eigenschaften zugewiesen werden, wodurch deren Bedeutung ausgedrückt werden kann.²

Auszeichnungssprachen dienen in erster Linie dazu, einem elektronischen Textdokument neben den Textdaten, die den Inhalt darstellen, zusätzliche Informationen hinzuzufügen³. Dafür wird der entsprechende Teil des Klartextes mit einer Start- und Endmarkierung versehen. Beispielsweise wird in HTML Text als Absatz ausgezeichnet, indem vor dem entsprechenden Teil `<p>` notiert wird. Um das Ende des Absatzes zu markieren, wird die Auszeichnung mit `</p>` beendet.

Mit **HTML** können neben dem bereits erwähnten Absatz beispielsweise auch Überschriften erzeugt werden, Tabellen, Listen oder Eigenschaften der Schrift wie Farbe, fett oder kursiv definiert werden. Hier liegt der Schwerpunkt auf der physischen Auszeichnung. Außerdem ist es mit HTML möglich, Grafiken oder andere multimediale Inhalte einzubinden und Verweise, beispielsweise auf andere Webseiten, zu setzen.

Zwar bietet HTML auch einige Gestaltungsmöglichkeiten für Dokumente, doch professionelle Webseiten kann man damit nicht gestalten. Daher gibt es die Möglichkeit, HTML durch die Formatsprache **CSS** (Cascading Style Sheets) zu ergänzen. Diese erlaubt das beliebige Formatieren einzelner HTML-Elemente. Der Trend geht dahin, in HTML überhaupt keine Formatierung mehr vorzunehmen, sondern nur noch durch CSS⁴.

Bei **XML** geht es vor allem darum, Texten semantische Eigenschaften zuzuweisen.⁵ Beispielsweise können Textelemente als Bestellmengen, Produktbezeichnungen oder Preise gekennzeichnet werden und diese in dem Dokument als Bestellungen zusam-

² Neumann 2011, o.S.

³ Loxen 1998, o.S., vgl. Rechenberg/ Pomberger 2006, S. 282

⁴ Vgl. Meyer 2005, S.8, Engels 2005, S.448f.

⁵ Vgl. Münz/ Nefzger 2002, S.79, Harold/ Means 2005, S.4

mengefasst werden. Mit XML kann aber nicht angegeben werden, wie die Darstellung erfolgen soll. XML beschreibt nur die Struktur des Dokuments.⁶

Zwar ist XML nicht ganz so einfach zu lernen wie HTML, doch es bietet zwei entscheidende Vorteile:

a) HTML besitzt vordefinierte Tags, mit denen der Inhalt des Dokuments sowie dessen Darstellung beschrieben wird, womit die Auszeichnungsmöglichkeiten begrenzt sind. In XML dagegen können und müssen sogar eigene Befehle definiert werden. Dadurch ist die Sprache erweiterbar (extensible).

b) HTML ist darauf ausgelegt, auch die Darstellung der Dokumentinhalte zu spezifizieren. Zum Beispiel wird mit dem `<h1>`-Tag eine Überschrift definiert und ihr eine bestimmte Größe zugewiesen. Es kommt zur Vermischung der Dokumentinhalte mit Darstellungsinformationen. Dies ist problematisch, da eine saubere Darstellung auf anderen Medien wie Web, Papier kaum möglich ist.⁷ In XML werden Struktur und Darstellungsinformationen strikt getrennt.

Zu Gestaltung von XML- Dokumenten wurde die Extensible Stylesheet Language (XSL) entwickelt. XSL besteht nicht nur aus Formateigenschaften, sondern „übernimmt auch erweiterte Aufgaben der Textverarbeitung wie das sortierte Ausgeben von listenartigen Daten, automatische Nummerierung usw.“⁸ Eine Alternative dazu ist CSS. Auch mit CSS lässt sich die Gestaltung von XML-Dokumenten realisieren.

XHTML (Extensible Hypertext Markup Language) bedeutet so viel wie erweiterbare Hypertext-Auszeichnungssprache. XHTML ist HTML, welches mit Hilfe von XML definiert wird anstatt wie bisher mit SGML⁹. Damit kein Versionenwirrwarr entsteht, entschloss man sich dazu, dieses neue, XML-basierte HTML mit einem neuen Namen und eigener Versionenkontrolle auszustatten.

Mit Auszeichnungssprachen lassen sich nicht nur Text-Dokumente strukturieren, sie lassen sich auch in einem plattformunabhängigen Dateiformat speichern, sodass ein Austausch leicht möglich ist. Zur Verdeutlichung des Problems: Text-Dokumente werden meist mit Anwendungen von Word oder Open Office erstellt und in deren herstel-

⁶ Vgl. Münz/ Nefzger 2002, S.79, Harold/Means 2005, S.4

⁷ Vgl. Born 2005, S.23, Vonhoegen 2009, S.30

⁸ Münz/ Nefzger 2002, S.80

⁹ SGML ist eine Meta-Sprache, die das Definieren von Auszeichnungssprachen mit Hilfe von Dokumenttyp-Definitionen (DTDs) erlaubt

lerspezifischen Formaten (.doc, .odf) gespeichert. Doch ein Austausch dieser Dokumente ist nicht möglich, wenn man nicht das geeignete Programm besitzt. Ein Konvertieren von einem zum anderen Format kann eventuell Fehler bergen. Dieses Problem wird durch Auszeichnungssprachen gelöst.¹⁰ Sowohl HTML-Dokumente als auch XML-Dokumente lassen sich in einem einfachen Text-Editor erstellen und jeder Browser kann sie lesen. Natürlich gibt es auch spezielle Editoren, die den Anwender beim Einfügen von Befehlen, Grafiken oder anderem unterstützen.

Die aktuellen Versionen von HTML und XML sind HTML 5 und XML 1.0. Doch HTML 5 ist noch nicht zum Standard erhoben worden, erst 2014 soll dies laut dem W3C geschehen¹¹.

¹⁰ Vgl. Born 2005, S.21

¹¹ W3C 2011, o.S.

3 Vorbetrachtungen

Vor der Konzeption des Komplexpraktikums ist es zunächst wichtig zu analysieren, für wen es genau konzipiert wird. Darum folgt nun eine kurze Zielgruppenanalyse. Anschließend werden die Inhalte für die Praktika festgelegt, wobei es später noch zu Veränderungen kommen kann, da sich erst bei der Umsetzung zeigt, inwiefern Inhalte gekürzt oder ergänzt werden müssen. Anschließend wird kurz festgehalten, wie das Komplexpraktikum aufgebaut sein wird.

Weiterhin soll auch die Didaktik diskutiert werden. Dies soll dabei helfen, die Lehrinhalte so zu gestalten, dass sie besser aufgenommen und nachhaltig verstanden werden.

3.1 Die Zielgruppe

Das Komplexpraktikum wird für Studenten aus dem Studiengang Energie- und Umweltmanagement konzipiert. Deshalb sollen Energie und Umwelt thematisch eine Rolle spielen. Somit haben die Studenten einen Bezug dazu und ihr Interesse ist gewährleistet.

Das Modul, in welchem das Komplexpraktikum stattfinden soll, liegt im ersten Semester. Das heißt, die Studenten kommen meist frisch aus der Schule und haben mit großer Wahrscheinlichkeit noch keine Berührung mit dem Thema Auszeichnungssprachen. Es kann davon ausgegangen werden, dass keine Informatikkenntnisse bestehen, außer denen, die in der Vorlesung vermittelt werden. Doch zumindest eine grundlegende Erfahrung mit Computern und dem Internet kann vorausgesetzt werden.

Die Schlussfolgerung daraus ist, dass eine schrittweise Einführung in das Thema Auszeichnungssprachen erfolgen muss. Das sehr komplexe Themenumfeld muss stark eingegrenzt sein. Letztendlich sollen die Studenten keine Informatiker werden, sondern nur ein Grundverständnis erlangen und einen Einblick in die Thematik bekommen.

Eventuell unterschiedliche Vorkenntnisse könnten durch flexible Aufgabenstellungen je nach Kenntnisstand oder Zusatzaufgaben ausgeglichen werden.

3.2 Methodische und didaktische Aspekte

Um Lehrinhalte möglichst nachhaltig zu vermitteln, ist es essentiell, eine Motivation zu schaffen. Im besten Fall sollen die Studenten sich nicht nur mit dem Thema befassen, weil sie es müssen, um das Studium zu schaffen. Um eine tiefere Motivation zu erreichen, gibt es verschiedene Möglichkeiten. Beispielsweise kann Lernmotivation durch Neugier erzeugt werden. Durch die Neugier entsteht das Bedürfnis, mehr Informationen über einen Gegenstand zu erfahren. Nach Winkel und Petermann ist die „Neugiermotivation“ eine „wichtige Antriebskraft des Lernens“. ¹²

Meistens ist die Neugier jedoch auf bestimmte Inhaltsbereiche beschränkt. Hierbei handelt es sich um Interessen. Der eine Schüler mag beispielsweise lieber Mathe, der andere Deutsch. ¹³ Das Interesse erhöht die Bereitschaft, sich mit einer Materie zu beschäftigen. ¹⁴ Daher wird auch das Thema Umwelt und Energie für das Komplexpraktikum gewählt. Die Studenten hätten den Studiengang Energie- und Umweltmanagement nicht gewählt, wenn das Thema nicht ihrem Interesse entsprechen würde.

Wichtig ist auch, auf vorhandenem Wissen aufzubauen. Erst, wenn das Gehirn den Lernstoff mit vorhandenem Wissen verknüpfen kann, wird er wirklich verinnerlicht. Nach Hüholdt lassen sich neue Lerngegenstände immer als Ergänzung oder Abänderung des bisher Gelernten begreifen. ¹⁵ Daher wird es auch eine Wiederholung und Festigung der Vorlesungsinhalte sowie kleine Übungen als Vorbereitung auf das Praktikum geben.

Für den Lernprozess ist es nach Hüholdt auch sinnvoll, Ergebnisse nicht als Fertiggericht anzubieten. Indem die Studenten durch Fragen, Übungen und das Praktikum aktiv involviert sind, werden sie am Prozess der Erkenntnisgewinnung beteiligt. Dies fördert das Verständnis. ¹⁶

Weiter wird das Lernen gefördert durch

¹² Vgl. Winkel/ Petermann/ Petermann 2006, S. 58f.

¹³ Vgl. Winkel/ Petermann/ Petermann 2006, S. 59f.

¹⁴ Vgl. Hüholdt 1993, S. 104, Herrmann 2006, S.11

¹⁵ Vgl. Hüholdt 1993, S. 106

¹⁶ Vgl. Hüholdt 1993, S. 376

a) Einsatz von z.B. Superlativen, Sensationen, Bizarrerien, Irritationen, Emotionalisierungen, Humor.

b) Förderung der Motivationselemente wie: Erfolgserlebnis, Interesse, Aufforderungscharakter, Faszinationscharakter, Vollendungswunsch¹⁷.

Wenn möglich sollten diese Elemente im Komplexpraktikum eingesetzt werden.

¹⁷ Hüholdt 1993, S. 376

3.3 Die Inhalte

Um das Thema Auszeichnungssprachen in ein Komplexpraktikum zu verpacken, werden die Auszeichnungssprachen HTML, XML, XHTML und HTML 5 behandelt. HTML und XML sind derzeit die gängigsten Auszeichnungssprachen. XHTML bietet sich an, da es HTML und XML praktisch in sich vereint. HTML 5 schließlich ist in aller Munde und wird in naher Zukunft (2014) laut W3C zum Web-Standard erhoben¹⁸. Weiterhin wird CSS Inhalt sein, denn CSS ist für die Darstellung von HTML unerlässlich. Auch in Kombination mit XML bietet es sich zu Darstellungszwecken an. Auch wenn sich dafür XML die Sprache XSL besser eignet, da diese eigens dafür entwickelt wurde, wird XSL für das Praktikum nicht eingesetzt. CSS reicht vollkommen zu Einführungszwecken in die Thematik aus.

Da der Zeitfond für 5 Veranstaltungen, in denen verschiedene Auszeichnungssprachen behandelt werden sollen, relativ klein ist und die Vorkenntnisse gering bis nicht vorhanden, werden diese Auszeichnungssprachen nur punktuell behandelt. Es werden Grundlagen vermittelt und verschiedene Anwendungs- und Einsatzmöglichkeiten aufgezeigt. Um jedoch auch im Thema Auszeichnungssprachen fortgeschrittene Studenten nicht zu unterfordern, werden Zusatzaufgaben bereitgestellt. Falls sich einzelne Praktika als zu kurz erweisen sollten, können die Zusatzaufgaben auch als generelle Aufgaben für alle gestellt werden. Um dies beurteilen zu können, muss das Komplexpraktikum aber erst einmal stattfinden.

3.4 Der Aufbau des Komplexpraktikums

Das Komplexpraktikum besteht aus fünf Veranstaltungen à 90 Minuten. Jedem Praktikum werden Vorbereitungsfragen und kleine Übungen vorausgehen, damit die Studenten die Vorlesungsinhalte verinnerlichen, sich mit den Themen eigenständig auseinandersetzen und sich gezielt auf das Praktikum vorbereiten können. Zu den Vorbereitungsaufgaben werden Literatur-Quellen angegeben, mit deren Hilfe die Studenten sich informieren können. Natürlich können sie zusätzlich eigene Recherche betreiben.

Inhalt eines jeden Praktikums wird jeweils eine komplexe Aufgabe sein. Durch eine komplexe Aufgabe statt mehrerer kleinen besteht nach Erfahrung des betreuenden

¹⁸ W3C 2011, o.S.

Professors eine größere Akzeptanz bei den Studenten. So haben sie am Ende eines Praktikums ein ansehnliches Endprodukt und der Zusammenhang vieler Dinge wird klarer. Die Praktikumsinhalte werden schließlich auf den Webbereich der Studenten, welcher von der Hochschule bereitgestellt wird, geladen. So können die Studenten ihre Arbeiten präsentieren, zum Beispiel vor der Familie.

Zu Beginn des ersten Praktikums werden die Studenten konkrete Vorgaben bekommen, wie sie mit Eclipse ihre Ordner-Struktur auf ihrem Computer einrichten sollen. So lernen sie, ihre Projekte sinnvoll und ordentlich zu strukturieren. Damit wird gewährleistet, dass nicht zu viel Zeit mit dem Suchen bestimmter Dateien vergeudet wird. Außerdem lernen die Studenten dadurch, korrekte Pfadangaben zu machen.

4 Konzeption des Komplexpraktikums

4.1 Schwerpunkte

Das Komplexpraktikum wird folgende Schwerpunkte behandeln:

Komplex I: HTML

Komplex II: HTML und CSS (Schwerpunkt CSS)

Komplex III: XML und CSS (Schwerpunkt XML)

Komplex IV: XML: XML-Schema und DTD

Komplex V: XHTML, HTML5, Zusammenfassung

In den folgenden Kapiteln werden die Inhalte und Ziele der einzelnen Komplexe erläutert.

4.1.1 Komplex I - HTML

Das erste Praktikum soll sich mit der Auszeichnungssprache HTML beschäftigen. Die Studenten sollen lernen, wie HTML-Dokumente erstellt werden. Weiter werden sie den Aufbau einer HTML-Datei kennenlernen, die Syntax und Regeln von Befehlen sowie Beispiele zur Textauszeichnung. Auch das Einbinden von Grafiken und das Setzen von Verweisen wird Inhalt sein sowie das Erstellen von Tabellen. Am Ende des Praktikums sollen die Studenten dazu in der Lage sein, eigene einfache HTML-Seiten zu erstellen, die die oben genannten Elemente beinhaltet.

Inhalte der Vorbereitung und des Praktikums:

- Workflow / das Erstellen von HTML-Dokumenten
- Aufbau / Grundgerüst
- Syntax, Regeln
- Beispiele zur Textauszeichnung
- Einbindung von Grafiken
- Verweise
- Tabellen

4.1.2 Komplex II - HTML und CSS

Im zweiten Praktikum soll das Formatieren von HTML-Webseiten mittels CSS Schwerpunkt sein. Die Studenten sollen ein Verständnis für CSS erlangen und punktuell einige Gestaltungsmöglichkeiten erlernen wie zum Beispiel durch Schriftformatierung, Farben oder Abstände. Mit diesem Handwerkszeug soll die im ersten Praktikum erstellte erste Webseite noch professioneller gestaltet werden können.

Inhalte der Vorbereitung und des Praktikums:

- Einbindung von CSS in HTML
- Syntax von CSS
- Schriftformatierung mit CSS
- Farben
- Formatierung von Verweisen (Links)
- Box-Modell, Abstände und Positionierung
- Eigene Klassen

4.1.3 Komplex III – XML und CSS

Im dritten Praktikum sollen die Studenten ihre Kenntnisse zum Aufbau eines XML- Dokuments sowie zur Syntax von XML vertiefen. Aufbauend auf ihren bereits erworbenen CSS-Kenntnissen werden Sie lernen, externe CSS-Dateien in XML einzubinden, um XML-Daten sinnvoll darstellen zu können. Hier wird insbesondere die zeilen- und spaltenweise Darstellung von Daten eine Rolle spielen.

Inhalte der Vorbereitung und des Praktikums:

- Aufbau eines XML-Dokuments
- Syntax, Regeln
- CSS mit XML verknüpfen
- Mit CSS Werte zeilen- und spaltenweise angeben

4.1.4 Komplex IV – DTD, XML-Schema

Das vierte Praktikum soll sich mit der Dokumenttyp-Definition sowie dem XML-Schema befassen. Die Studenten sollen eine eigene DTD erstellen und in XML einbinden sowie ein XML-Schema grafisch umsetzen und ebenso in eine XML-Datei einbinden. Anhand dessen werden sie die Unterschiede beider Methoden lernen und sich mit Datentypen beschäftigen.

Inhalte der Vorbereitung und des Praktikums:

- Erstellung einer DTD und eines XML-Schema
- Einbinden einer DTD und eines XML-Schema
- Datentypen
- Häufigkeiten
- Vorteile des XML-Schema

4.1.5 Komplex V – XHTML, HTML5, Zusammenfassung

Das fünfte und letzte Praktikum soll in XHTML einführen und die wesentlichen Unterschiede zu HTML verdeutlichen. Die Studenten sollen mit der Syntax von XHTML vertraut werden und dazu befähigt, eigene Dokumente in XHTML zu erstellen. Weiterhin soll HTML 5 beispielhaft vorgestellt werden. Zum Abschluss sollen die Studenten noch einmal einen Überblick über die verschiedenen Auszeichnungssprachen bekommen, welche Standards für welche Zwecke geeignet sind und was deren wichtigsten Merkmale sind.

Inhalte der Vorbereitung und des Praktikums:

- Regeln, Unterschiede zu HTML
- HTML- in XHTML-Dokumente umschreiben
- HTML5 - beispielhaft
- Vergleich/Gegenüberstellung Auszeichnungssprachen: HTML – XHTML – XML

5 Umsetzung des Komplexpraktikums

Nach einer groben Festlegung der Inhalte für das Komplexpraktikum, wurden konkrete Projekte realisiert (siehe CD), durch welche die Inhalte des vorigen Kapitels zur praktischen Anwendung kommen. Für Komplex 1 entstanden zwei reine HTML-Seiten und für Komplex 2 ein Stylesheet für diese HTML-Seiten, welches in HTML eingebunden wurde. Zur Anwendung von XML wurde in Komplex 3 eine XML-basierte Tabelle mit Hilfe von CSS erstellt. Für Komplex 4 entstanden eine DTD sowie ein XML-Schema für eben diese Tabelle. Komplex 5 beinhaltet die als XHTML umgeschriebenen HTML-Dateien aus Komplex 1 und 2. Zur Demonstration von HTML5 wurde zusätzlich ein Video in die Webseite eingebunden. Als letztes Element für Komplex 5 entstand eine auf XML basierende Tabelle. Mit Hilfe dieser findet ein Vergleich zwischen HTML, XML und XHTML statt. Die umgesetzten Projekte sind auf der beiliegenden CD zu finden.

5.1 Verwendete Technik

Die Konzeption und Umsetzung der Praktikums-Komplexe erfolgte unter Windows XP (32 Bit) und mit den Web-Browsern Internet Explorer 8 und Mozilla Firefox 12.

Im Praktikum wird als Entwicklungsumgebung Eclipse (IDE for Java EE Developers¹⁹) eingesetzt. Die Nutzung dieses Programms bietet sich an, da mit ihr sich die verschiedensten Dokumente erstellen lassen, unter anderem auch XML-Dokumente, DTDs und XML-Schemata. XML-Schemata lassen sich sogar grafisch realisieren. Auch für den Java-Unterricht werden die Studenten Eclipse nutzen und müssen ohnehin damit umgehen können.

Die Arbeitsplätze im Praktikum sind mit Computern ausgestattet, auf denen Eclipse vorhanden ist. Weiterhin muss Java installiert sein²⁰.

Parallel zu der Umsetzung der Projekte wurden die Vorbereitungsfragen erstellt, die zur Lösung der Praktikumsaufgabe helfen sollen. Außerdem wurden Literaturquellen angegeben, die auch bei der Erstellung der Projekte hilfreich waren.

¹⁹ Eclipse kann heruntergeladen werden unter: <http://www.eclipse.org/downloads/>

²⁰ eine Anleitung zur Installation von Java befindet sich unter :
http://www.java.com/de/download/help/download_options.xml

Technische Voraussetzung für das Komplexpraktikum ist ein Computer mit Windows XP, worauf die Software Eclipse (IDE for Java EE Developers) vorhanden und Java installiert ist.

Es folgt nun eine Dokumentation der konkreten Umsetzung der einzelnen Komplexe.

5.2 Komplex I

Der erste Komplex hat den Schwerpunkt HTML, zunächst ohne CSS. Am Ende dieses Komplexes sollen die Studenten in der Lage sein, eine eigene HTML-Seite erstellen zu können.

Als Literaturquellen für dieses Kapitel wurde hauptsächlich die Seite SelfHTML von Stefan Münz benutzt, welche auch als Buch erschienen ist²¹. Die Web-Seite behandelt umfassend HTML und CSS und wird stetig aktualisiert. Zusätzlich wurde eine Website von Peter Will, Professor der HS Mittweida, benutzt²². Dort ist unter anderem Zusatzmaterial zur Vorlesung zu HTML und CSS zu finden. Weitere Literatur zu HTML und den theoretischen Grundlagen sind beispielsweise die Bücher „Die Sprachen des Web: HTML und XHTML“ von Tolksdorf²³ und „HTML/XML & JavaScript“ von Hanke²⁴. Aber auch im Internet gibt es zahlreiche Webseiten, die sich mit HTML beschäftigen.

5.2.1 Die Vorbereitung

Durch die Vorbereitungsfragen sollen die Studenten alle nötigen Kenntnisse für das Praktikum erwerben. Zunächst sollen die Studenten verinnerlichen, was HTML bedeutet. HTML (hypertext markup language²⁵) ist eine textbasierte Auszeichnungssprache zur Strukturierung von Inhalten wie Texten, Bildern und Hyperlinks in Dokumenten. Mit HTML lässt sich Text, zum Beispiel als fett oder kursiv, auszeichnen. Aber HTML bietet noch mehr Möglichkeiten, Text zu strukturieren. Auch das Referenzieren von Bildern ist möglich. Und mit den Hyperlinks (Verweisen) bildet HTML die Grundlage des Webs. Ohne die Hyperlinks wäre das Wechseln von einer Seite zur nächsten im Web nicht möglich.

Weiter sollen die Studenten wissen, dass HTML mit einfachen Texteditoren geschrieben werden können, aber auch mit komplexeren Entwicklungsumgebungen wie Eclipse, dem Phase HTML-Editor oder weiteren. Diese Entwicklungsumgebungen unterstützen den Anwender zum Beispiel durch farbliche Hervorhebung von HTML-Elementen oder durch einfaches Einfügen bestimmter Elemente oder Definitionen.

²¹ Münz/ Nefzger 2002, o.S.

²² Will o.J., o.S.

²³ Tolksdorf 2000, o.S.

²⁴ Hanke 2000, o.S.

²⁵ zu Deutsch: Hypertext-Auszeichnungssprache

Wichtig ist auch, das Grundgerüst von HTML zu kennen. Es ist die Grundlage jeder HTML-Datei. Darum sollen die Studenten in der Vorbereitung die drei Bereiche des Grundgerüsts (Dokumenttyp-Deklaration, Head, Body) nennen.

Weiter geht es in der Vorbereitung mit Werkzeugen zur Textstrukturierung: Überschriften, Absätzen und Zeilenumbrüchen. Diese und weitere Elemente geben einem Dokument seine Struktur. Durch sie ist der Text nicht nur Text, sondern er ist als Überschrift, Absatz oder anderes erkennbar. Daher sollen die Studenten diese Elemente recherchieren. Dabei müssen sie erst einmal wissen, was ein Tag ist. HTML-Elemente werden durch so genannte „Tag“s markiert. Fast alle HTML-Elemente werden durch ein einleitendes und ein abschließendes Tag markiert. Tags werden in spitzen Klammern notiert: < >

Anschließend sollen auch einige Elemente zur physischen Textauszeichnung erlernt werden. Zwar werden diese zukünftig durch CSS realisiert werden, aber dennoch sollten die Studenten sie einmal benutzt haben.

Als nächstes sollen die Studenten herausfinden, was „Standalone-Tags“ sind. Es handelt sich dabei um Elemente, die keine schließenden Tags besitzen. Dies gibt es nur bei HTML, nicht bei XML und XHTML. Um später die Unterschiede zwischen HTML und XML/XHTML deutlich zu machen, ist dieses Vorwissen nützlich.

Weiter wird in der Vorbereitung abgefragt, wie Grafiken und Tabellen eingefügt werden. Die Studenten sollen die Befehle kennen, um sie später im Praktikum anwenden zu können. Grafiken werden in HTML über eine Referenz eingebunden. Beim Referenzieren von Bildern ist zu beachten, dass der korrekte Pfad angegeben wird. Es muss ein relativer Pfad angegeben werden. Hierbei liegt das Bild entweder im selben Ordner wie die HTML-Datei oder in einem extra Ordner im selben Verzeichnis.

Bild im gleichen Verzeichnis ``

Bild in einem Ordner „Bilder“,
der sich im Verzeichnis befindet ``

Desweiteren kann beim Einbinden der Grafik ein Alternativtext angegeben werden. Dieser erscheint, wenn das Bild vom Browser nicht angezeigt werden kann. Aber auch Menschen, die die Webseiten nicht lesen können und sie deshalb akustisch wahrnehmen, können dadurch trotzdem den Inhalt des Bildes erfahren. Natürlich ist immer ab-

zuwägen, wann dies notwendig ist. Wenn das Bild nicht inhaltlich relevant ist, sondern nur schmückendes Beiwerk, kann darauf verzichtet werden, um den Textfluss nicht zu unterbrechen.²⁶ Es sollte sich immer bemüht werden, die Webseiten so zu gestalten, dass jeder sie nutzen und lesen kann. Auch ist es sinnvoll, die Breite und Höhe des Bildes anzugeben, damit der Platz für das Bild beim Laden der Seite reserviert wird.

Anschließend soll recherchiert werden, wie Verweise (Hyperlinks) gesetzt werden. Hier gibt es mehrere Möglichkeiten. Zum einen kann auf Dateien innerhalb des World Wide Web über eine URL verwiesen werden. Eine URL ist eine Adressierungsform für Internetadressen²⁷, zum Beispiel <http://www.google.de>. Zum anderen ist es möglich, über einen sogenannten „Anker“ auf eine bestimmte Textposition zu verweisen. Das wird zum Beispiel genutzt, um vom unteren Ende der Seite wieder an eine Textstelle am Anfang der Seite zu verweisen. Die dritte Möglichkeit ist, zu einer HTML-Seite innerhalb des Projekts zu referenzieren. Meist besteht eine Website aus mehreren Dokumenten, die gegenseitig aufeinander verweisen.

Zum Schluss wird hinterfragt, wie sinnvoll es ist, in HTML das Seiten-Layout zu definieren. Der Hintergrund ist, dass die Studenten in HTML den Text möglichst nur strukturieren sollen und nicht gestalten oder formatieren. Denn dafür sollte CSS genutzt werden. CSS bietet viel mehr Möglichkeiten zur Gestaltung als HTML und ist dafür besser geeignet. Wenn das Layout in HTML definiert wird, bedeutet das beispielsweise, dass der HTML-Quelltext sehr unübersichtlich und unnötig aufgeblasen wird, was auch längere Ladezeiten zur Folge hat. Außerdem können Bereiche unabhängig von Zeilen oder Spalten definiert werden, was - je nach Struktur der Seite - oft ein Vorteil ist. Eine große Stärke von CSS ist auch, dass sich über eine externe CSS-Datei ein Layout auf verschiedene Seiten parallel übertragen lässt. In HTML müsste für jede Seite das Layout neu definiert werden. Deshalb hat sich im Webdesign die Trennung von Struktur und Layout durchgesetzt. In HTML wird die Struktur der Webseite (Überschriften, Absätze usw.) definiert und CSS übernimmt die Darstellungsinformationen.²⁸

In einer kleinen Übung sollen die Studenten dann eine HTML-Seite erstellen, die nur aus dem Grundgerüst besteht sowie einem eigenen kleinen Text. Hierdurch sollen sie durch aktives Ausprobieren den Workflow des Erstellens einer HTML-Datei erlernen und merken, wie einfach das geht.

²⁶ Vgl. Radtke/ Charlier 2006, S.86f.

²⁷ DATACOM 2012, o.S.

²⁸ Vgl. Meyer 2005, S.3-6

5.2.2 Das Praktikum

Aufgabe des Praktikums ist es, zwei HTML-Seiten zu erstellen, welche bestimmte Elemente beinhalten. Es sind zwei, damit einerseits die Seiten untereinander verlinkt werden können und damit das Prinzip der Verwendbarkeit einer CSS-Datei für mehrere Seiten deutlich wird.

Die Studenten sollen das, was sie während der Beantwortung der Vorbereitungsfragen gelernt haben, anwenden. Dabei sind die Vorgaben relativ allgemein gehalten. Welche genauen Werte verwendet werden, bleibt den Studenten überlassen. So können sie etwas kreativ sein oder auch verschiedene Werte ausprobieren und nach dem Anschauen im Browser entscheiden, wie sie die Vorgaben genau umsetzen möchten.

Im gesamten Komplexpraktikum soll das Thema Energie und Umwelt eine Rolle spielen. Darum basiert die erste HTML-Seite auf einem Text, in dem darum geht, dass die Informations- und Kommunikationstechnik grüner werden sollte, um die Umwelt zu schonen²⁹. Das eingebundene Bild zeigt einen Stromstecker, der vor einer Steckdose zu sehen ist. Das Motiv passt gut in den Kontext. Das blaue Licht im Bild verdeutlicht eine gewisse Gefahr, die von der Stromnutzung ausgeht.



Abbildung 1 Ausschnitt der HTML Seite (1)

Auf der zweiten Seite soll eine Tabelle eingebunden werden und diese mit Daten zum Stromverbrauch des Internets gefüttert werden. Das Thema Stromverbrauch und Internet wird hiermit wieder aufgegriffen. Die Daten stammen aus einer Übersicht (siehe Abb. 2), welche in der Zeitschrift „neue energie“ zu finden war.

²⁹ Behrendt/ Erdmann 2009, S.4,6

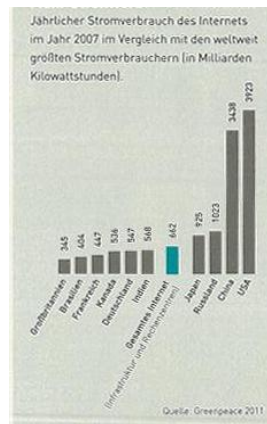


Abbildung 2 Übersicht über jährlichen Stromverbrauch der größten Stromverbraucher-Länder in Bezug auf das Internet³⁰

5.3 Komplex II

Im zweiten Praktikum sollen die HTML-Seiten aus Komplex 1 mit einem Style Sheet gestaltet werden. Dazu bekommen Sie konkrete Vorgaben, wie die Umsetzung geschehen soll. Durch die Vorbereitungsfragen sind die Studenten dazu aufgefordert, sich mit CSS in seinen Grundzügen und sich mit einigen praktikumsrelevanten Anwendungsbeispielen auseinanderzusetzen.

Außerdem gibt es eine extra Aufgabe für Studenten mit fortgeschrittenen HTML- und CSS- Kenntnissen. HTML und CSS sind im Web heutzutage ständige Begleiter. Viele junge Leute haben eine eigene Homepage oder nutzen Content Management- Systeme. Daher ist es nicht ganz unwahrscheinlich, dass hier einige Studenten Kenntnisse mitbringen. Damit diese sich nicht unterfordert fühlen, bekommen sie die Zusatzaufgabe, die HTML-Seiten weiter optisch ansprechend zu gestalten.

Literaturquellen für dieses Kapitel sind die Website SelfHTML³¹ von Stefan Münz und zudem die Website css4you³² - eine sehr umfangreiche deutsche CSS-Referenz.

³⁰ Wiesen 2011, S.32

³¹ Münz/ Nefzger 2002, o.S.

³² css4you 2009, o.S.

5.3.1 Die Vorbereitung

Durch die Vorlesung und das Lösen der Vorbereitungsfragen sollten die Studenten wissen, was Cascading Style Sheets sind: Formatvorlagen für Webseiten. CSS ist eine Ergänzungssprache, mit der HTML- oder auch XML- Elemente exakt formatiert und positioniert werden können. Um einen Eindruck von dieser Sprache zu bekommen, sollen die Studenten weiter angeben, wie CSS-Elemente aussehen. Sie sollten wissen, dass ein CSS-Befehl aus dem sogenannten Selektor und der Stylesheet-Angabe (Deklaration) in geschweiften Klammern besteht.

```
Selektor {Deklaration}, z.B.: h1{color:red;}
```

Auch die Einbindung in HTML sollte recherchiert werden, um später die CSS-Datei in die HTML-Seiten einbinden zu können. Die Studenten sollten wissen, dass CSS sowohl direkt in HTML in den Head-Bereich integriert als auch als externe Datei eingebunden werden kann. In der Aufgabe wird eine externe Datei verlangt, um den großen Vorteil darin zu zeigen, ein Stylesheet für alle Seiten eines Web-Projekts verwenden zu können. Dadurch kann ein einheitliches Erscheinungsbild eines Web-Projekts geschaffen werden. Für die Einbindung als externe Datei wird eine Textdatei benötigt, die die Stylesheet-Angaben enthält und die Endung `.css` hat. In der HTML-Datei wird durch das `<link>`-Tag im Head-Bereich auf diese externe Datei verwiesen. Dies kann absolut oder relativ geschehen. Für eine absolute Referenz wird im Attribut `href=` Pfad und Name der Datei angegeben. Liegt diese Datei auf einem anderen Server, kann auch die komplette URL-Adresse angegeben werden. Besser als die absolute ist allerdings eine relative Pfadangabe:

```
<link rel="stylesheet" type="text/css" href="dateiname.css">
```

In obigem Beispiel liegt die CSS-Datei im selben Ordner wie die HTML-Datei. Läge die CSS-Datei in einem extra Ordner im gleichen Verzeichnis, müsste da stehen `href=Ordner/dateiname.css`. Der Vorteil der relativen Pfadangabe, ist dass das Webprojekt problemlos auf einen anderen Server verschoben werden kann, ohne den Pfad zu ändern.

Weiter sollen sich die Studenten damit auseinandersetzen, wie Schriftfamilien definiert werden. Schriftfamilien sind beispielsweise Arial, Helvetica, Times Roman usw. Es ist möglich, mehrere Schriftfamilien hintereinander anzugeben. Wichtig ist nur, dass die Werte durch ein Komma voneinander getrennt sind. Der Browser zeigt dann die Schrift an, die auf dem Betriebssystem installiert ist. Außerdem müssen Schriftarten und Schriftfamilien, die ein Leerzeichen beinhalten, in Anführungszeichen gesetzt werden ("Arial Black" oder 'Arial Black').

Beispiel: `p {font-family:verdana, arial, sans-serif;}`

Auch Farben sind Bestandteil jedes Web-Dokuments. Daher sollen die Studenten wissen, wie Farbangaben gemacht werden. Grundsätzlich gibt es zwei Möglichkeiten, Farben in HTML zu definieren: entweder durch Angabe der RGB-Werte der gewünschten Farbe in Hexadezimalform oder durch Angabe eines Farbnamens.

Hexadezimale Farbangaben werden nach dem Schema #RRGGBB angegeben; R ist der Rotwert, G der Grünwert und B der Blauwert der Farbe. Eine Hexadezimalzahl kann aus den Zahlen 0-9 und den Buchstaben A-F bestehen. Sie kann also 16 mögliche Werte annehmen. Mit einer zweistelligen Hexadezimalzahl lassen sich je Grundfarbe also $16 \times 16 = 256$ Werte von 0 bis 255 darstellen. Die sechsstellige Hexadezimalzahl bildet den gesamten RGB-Farbraum ab, also $256 \times 256 \times 256 = \text{ca. } 16,7$ Millionen Farben. Für die Ausgabe am Monitor kommen nur RGB-Werte in Frage.

Weiterhin ist das Definieren von Breite und Höhe eines Elements Bestandteil der Vorbereitung. Bei der Angabe von Längeneinheiten muss abgewogen werden, ob absolute oder relative Maßangaben verwendet werden. Nicht alle Bildschirme haben die gleiche Größe und das gleiche Format. Auch die Pixeldichte kann von Bildschirm zu Bildschirm verschieden sein. Darum ist es für digitale Dokumente manchmal besser, relative Längenangaben zu benutzen, anstatt zum Beispiel pixelgenaue Angaben.

Als Beispiel wurde in folgende HTML-Seite ein Bild eingefügt mit 300 Pixeln Abstand vom linken Rand. Bei einer Auflösung von 800x600 Pixeln (siehe Abb.3) ist das Bild somit mittig. Bei einer Auflösung von 1400x1050 (siehe Abb.4) ist das Bild viel weiter links und zudem kleiner.

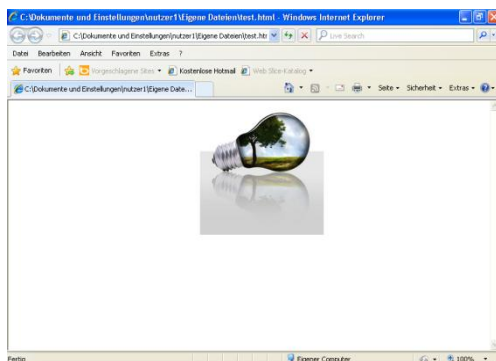


Abbildung 3 Auflösung 800x600

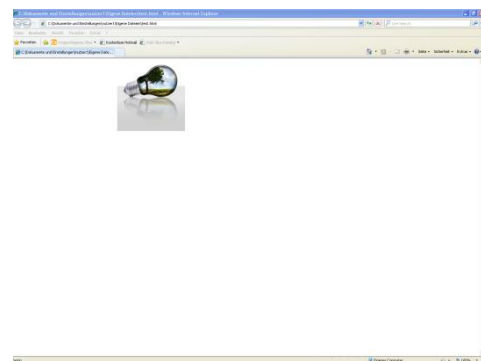


Abbildung 4 Auflösung 1400x1050

Absolute Längenangaben sind zum Beispiel Zentimeter (cm), Inches (in) oder Punkt (pt). 1 Punkt entspricht 1/72 Inches (Zoll). Relative Maßangaben sind Prozentangaben, „em“ und „ex“. Em steht für „bezogen auf die Schriftgröße des Elements“. Wenn bei-

spielsweise eine Überschrift mit 18 px festgelegt ist und diese einen Abstand von 1em bekommt, ist dieser Abstand ebenfalls 18 Pixel groß. 2 em entsprächen dann 36 Pixeln usw.

```
h1 {font-size: 18px;}  
h1 {margin-left: 1em;}
```

Die Einheit „xm“ drückt aus, dass die Größe abhängig von der Höhe eines kleingeschriebenen x der verwendeten Schrift ist. Diese Einheit wird aber nicht oft benutzt.³³ Selbst die Angabe in Pixeln gilt in dem Sinne als relative Maßeinheit, da die Darstellung abhängig ist von der Pixeldichte des Ausgabegerätes. Zum Beispiel hat ein Notebook eine andere Pixeldichte als beispielsweise ein TFT-ein Monitor. Auf ein Ausgabegerät bezogen ist sie jedoch auch absolut.

Weil es so viele verschiedene Ausgabegeräte gibt, ist es schwierig, das Layout einer Webseite so zu gestalten, dass sie bei allen Nutzern gleich aussieht. Münz und Nefzger empfehlen daher: „Vermeiden Sie deshalb vor allem bei Angaben, die für die Leserlichkeit von Text relevant sind, extreme Werte, und reagieren Sie, wenn Sie von mehreren Stellen zu hören bekommen, Ihre Seiten seien schwer lesbar.“³⁴ Welche Maßeinheit geeignet ist, muss immer im Einzelfall entschieden werden.

Als nächstes sollen sich die Studenten mit der Gestaltung von Verweisen vertraut machen. Der Browser stellt Verweise standardmäßig blau (noch nicht angeklickt) oder violett (schon einmal angeklickt) und außerdem unterstrichen dar. In HTML oder CSS können die Farben verändert werden. Eine Vermeidung des Unterstreichens ist jedoch nur über CSS möglich. Dazu wird die CSS-Angabe `text-decoration:none` benutzt.

Um den Textfluss zu steuern, kann die Eigenschaft „float“ benutzt werden. Sie wird im Praktikum dafür benötigt, den Text rechts um das Bild fließen zu lassen. Darum sollen die Studenten recherchieren, was es mit „float“ auf sich hat. Außerdem soll die Tabelle im Praktikum mit einem Rahmen versehen werden. Dazu sollen vorbereitend verschiedene Rahmentypen aufgezählt werden. Mit CSS lassen sich Rahmentypen wie gepunktet, gestrichelt, durchgezogen oder als 3D-Effekt (siehe Abb. 5) definieren. Genauso lässt sich festlegen, dass kein Rahmen erscheinen soll.

³³ w3C 2012, o.S.

³⁴ Münz/ Nefzger 2002, S.428

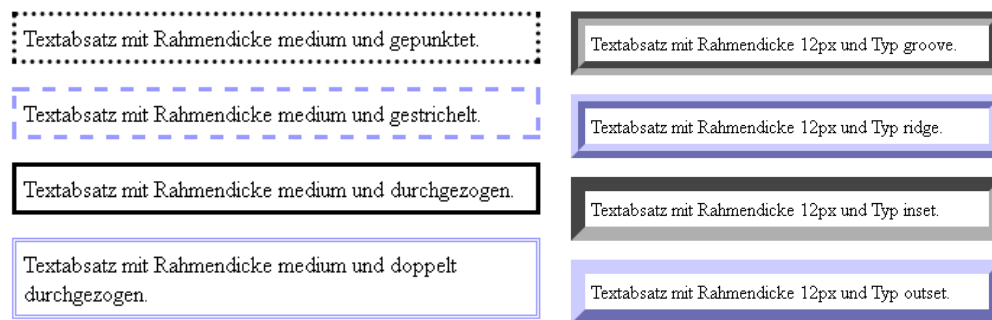


Abbildung 5 Rahmentypen mit CSS

Anschließend sollen sich die Studenten mit dem Definieren von Klassen mit den Elementen `div` und `span` beschäftigen. Klassen sind wie in der Programmierung eine Art Vorlage. Elemente, die die Klasse verwenden, erben ihre Eigenschaften. Die beiden HTML-Elemente `div` und `span` haben besondere Bedeutung für Stylesheets weil sie selbst relativ eigenschaftslos sind. Der einzige Unterschied besteht darin, dass das `div`-Element eine neue Zeile im Textfluss erzwingt, während `span` innerhalb eines Textes verwendet werden kann und keinen neuen Absatz erzeugt.

Um eine `div`-Klasse zu erstellen, muss der auszuzeichnende Inhalt zwischen einem öffnenden und schließenden `div`-Tag stehen. Hinter `div` folgt `class` mit dem Klassennamen, welcher in der CSS-Datei als Selektor dient. Beim `span`-Tag gilt dies analog.

```
<div class="name">
Inhalt
</div>
```

In der CSS-Datei steht als Selektor der Klassenname mit einem Punkt davor.

```
.name{font-family:"Arial Black"; color: #3e3e3e;}
```

Ein weiterer wichtiger Aspekt von CSS ist das Boxmodell. Es besagt, dass sich die Gesamtbreite eines Elements aus einer Addition der Breite des Elementinhalts, des Innenabstands, der Rahmenstärke sowie des Außenabstands errechnet. Das Definieren von Außen- und Innenabständen im Zusammenhang mit dem Boxmodell bietet viel Gestaltungsspielraum. Es ist auch möglich, alle vier Seiten eines Elements separat zu formatieren. In der Vorbereitung sollen sich die Studenten damit auseinandersetzen und im Praktikum ihre eigenen Erfahrungen machen.

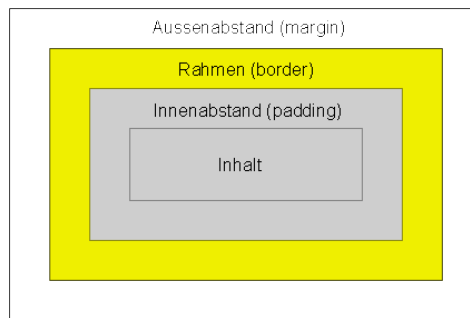


Abbildung 6 Box-Modell³⁵

5.3.2 Das Praktikum

Im Praktikum werden die Studenten eine externe CSS-Datei erstellen und sie mit den HTML-Dateien aus Komplex 1 verbinden. Dabei gibt es kleine Teilaufgaben mit Anforderungen, was das Stylesheet enthalten soll.

Die Aufgabe im Praktikum gibt meist grobe Vorgaben, sodass die Studenten frei in ihren Gestaltungsmöglichkeiten sind. Zum Beispiel können Farben frei gewählt werden. Nur bei Abständen werden genaue Pixelvorgaben gemacht, damit gesichert ist, dass die Abstände auch so groß sind, dass sie sichtbar sind. Außerdem sollen die Studenten erst einmal ein Gefühl für Pixelmaße bekommen. Ansonsten können sich die Studenten an den beigefügten Screenshots orientieren oder eigene Ideen einbringen. Hier könnte auch eine gute Motivation darin liegen, die eigenen Seiten noch schöner/besser gestalten zu wollen. Ansonsten sollte es Motivation genug sein, eine schmucklose HTML-Seite mit CSS zu verschönern.

Als Zusatzaufgabe für fortgeschrittene HTML- und CSS-Anwender gilt es, relativ eigenständig die HTML-Seiten zu gestalten. Dabei liegt der Fokus auf der Definition eigener Klassen und Individualformate sowie der Positionierung. Dabei muss die Angabe von absoluter und relativer Positionierung beachtet werden.

³⁵ css4you 2009, o.S.

5.4 Komplex III

In Komplex drei werden sich die Studenten mit XML beschäftigen. Aufgabe im Praktikum wird es sein, eine XML-Struktur einer Tabelle zu erstellen und diese mit CSS zu formatieren. Grundlage für die Tabelle sind die Daten, aus denen in den vorangegangenen Komplexen schon eine Tabelle entstanden ist. Nur diesmal wird sie in XML statt in HTML umgesetzt.

Als Literaturquellen für dieses Kapitel dienen hauptsächlich das Buch „Jetzt lerne ich XML“ von Günther Born³⁶ und die Website w3schools³⁷ - eine sehr große englischsprachige Webentwicklungsseite mit Referenzen und Anleitungen für viele Websprachen.

5.4.1 Die Vorbereitung

Durch die Vorbereitung sollen sich die Studenten mit XML und seiner Bedeutung beschäftigen. Sie sollten wissen, dass XML ebenso wie HTML eine Auszeichnungssprache ist. Der größte Unterschied zu HTML besteht darin, dass in XML eigene Tags definiert werden. Darum heißt sie auch erweiterbar. XML ist sehr flexibel einsetzbar. Es kann zur Beschreibung von Strukturen beliebiger Fachgebiete genutzt werden, egal ob in der Mathematik, in der Literatur oder im Bereich Multimedia.

Weiter sollten die Studenten sich mit dem Aufbau von gültigen XML-Dokumenten und mit den Regeln in XML vertraut machen. Dazu soll auch die erste Übung (siehe Abb. 7) dienen. Beispielsweise müssen alle Elemente ein schließendes Tag haben. Nur leere Elemente wie `br` enden mit Leerzeichen und „/`>`“.

```
<p>...</p>  
<br />
```

Außerdem muss bei XML im Gegensatz zu HTML die Groß- und Kleinschreibweise beachtet werden. Wird ein Element einmal großgeschrieben, muss es im Folgenden immer großgeschrieben werden.

Richtig:	<code><person></person></code>
Falsch:	<code><person></Person></code>

³⁶ Born 2005, o.S.

³⁷ w3schools o.J., o.S.

Es gibt noch einige Regeln mehr. Wenn sie alle bei der Erstellung eines XML-Dokuments befolgt werden, gilt es als „wohlgeformt“. Zu den Regeln gibt es in der Vorbereitung eine erste Übung (siehe Abb. 7), um das Gelernte anzuwenden und dadurch zu festigen. In der ersten Übung sollen Fehler gefunden werden. Jedes Beispiel enthält verschiedene Fehler.

Übung: Finden Sie in den folgenden XML-Beispielen die Fehler!

a)	b)
<pre><person> <vorname>Julia</vorname> <Nachname>Schmidt</nachname> <alter>22</alter> </Person></pre>	<pre><vorname>Julia</vorname> <nachname>Schmidt</nachname> <alter>22</alter></pre>
c)	d)
<pre><person> <vorname>Julia <nachname>Schmidt <alter>22 </person></pre>	<pre><i>Dieser Text ist fett und kursiv</i></pre>
	e)
	<pre><leeres_element></pre>
	f)
	<pre><person geschlecht=weiblich></pre>

Abbildung 7 Übung zu XML-Regeln

Im Beispiel a) sollte herausgefunden werden, dass bei XML die Groß- und Kleinschreibung beachtet werden muss. Wenn ein Tag einmal kleingeschrieben wurde, muss diese Schreibweise beibehalten werden. Bei Beispiel b) fehlt das Wurzelement. Die Studenten sollten wissen, dass ein Wurzelement zwingend nötig ist bei XML. Beispiel c) soll testen, ob die Studenten wissen, dass jedes Tag geschlossen werden muss. Es sei denn, es ist ein leeres Element. Leere Elemente werden besonders gekennzeichnet, was in Beispiel e) abgefragt wird. In Beispiel d) kommt es auf die korrekte Verschachtelung von Elementen an, bei f) muss beachtet werden, dass Attribute immer in Anführungszeichen stehen müssen.

Die nächste Vorbereitungsfrage beschäftigt sich mit dem Aufbau gültiger XML-Dateien. XML-Dateien sind erst gültig, wenn eine Grammatik (DTD oder XML-Schema) zugrunde liegt. Die DTD und das XML-Schema geben die Regeln vor, wie das XML-Dokument strukturiert sein soll. Sie definieren beispielsweise, wie viele Elemente oder Attribute vorhanden sein dürfen oder in welcher Reihenfolge.

Optional kann eine XML-Deklaration angegeben werden. Die XML-Deklaration ist eine Auszeichnung, die ausdrückt, dass es sich bei dem Dokument um eine XML-Datei handelt:

```
<?xml version="1.0"?>
```

Die XML-Deklaration kann außer der benutzten XML-Version auch zwei weitere Attribute enthalten: eines zur verwendeten Zeichenkodierung, und ein Ja/Nein-Attribut, ob die zugehörige DTD in der aktuellen Datei steht oder in einer separaten Datei.

Beispiel:

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="yes"?>
```

Die Zeichenkodierung spielt eine Rolle, wenn die Datei besondere Zeichen enthält, die im Englischen nicht gebräuchlich sind, zum Beispiel Umlaute, Zeichen aus anderen Sprachen oder Sonderzeichen. Damit die Zeichen korrekt auf dem Bildschirm angezeigt werden können, kann die Angabe einer bestimmten Zeichencodierung nötig sein. Zum Beispiel gibt es Zeichencodierungen für westeuropäische, hebräische oder für türkische Sprachen. Dateien, die in einem Computer gespeichert sind, bestehen im Prinzip aus Nullen und Einsen (Binärcode). Damit aus diesem Binärcode lesbare Zeichen werden, die sich am Bildschirm darstellen lassen, braucht es eine Konvention, welches Zeichen mit welchen Byte-Werten gespeichert wird. Diese Aufgabe haben die so genannten Zeichensätze. Nach Münz und Nefzger sind Zeichensätze „Tabellen, die einem Byte-Wert ein Zeichen zuordnen, das in unseren Schriftkulturen eine Bedeutung hat“³⁸. Eine Zeichencodierung „stellt die Zuordnungen zwischen den Bytes, die im Computer für Zahlen stehen, und den Zeichen im Zeichensatz her“³⁹. Es gibt verschiedene Zeichensätze, also Vorräte an Elementen (Zeichen), zum Beispiel:

- ASCII
- UTF-8
- ISO-8859-1

Laut Ishida vom W3C ist UTF-8 „die beste Empfehlung“ für eine Zeichencodierung, wenn Inhalte ins Web gebracht werden sollen⁴⁰.

Neben der Nutzung von Zeichensätzen und -kodierungen können Zeichen auch durch sogenannte Entities⁴¹ ersetzt werden. Entities (Singular: Entity) sind definierte Kürzel. Zum Beispiel müssen Zeichen wie „<“ oder „>“ durch einen Platzhalter ersetzt werden, da diese Zeichen eigentlich zur Kennzeichnung der Tags vorgesehen sind. Sie werden andererseits benötigt, um für längere Ausdrücke, die in den Anwendungsdaten einer

³⁸ Münz/ Nefzger 2002, S.1093

³⁹ Ishida 2011, o.S.

⁴⁰ Ishida 2011, o.S.

⁴¹ zu Deutsch: Entitäten

XML-Struktur immer wieder vorkommen, wie etwa Eigennamen oder Floskeln, in der DTD Kürzel zu definieren.⁴²Für den langen Ausdruck kann dann einfach das Kürzel benutzt werden und dies so oft wie nötig.

Die Studenten sollen sich mit Entities sowie mit den Zeichensätzen- und Codierungen vertraut machen. Dadurch können im Praktikum die Umlaute auf dem Bildschirm korrekt angezeigt werden.

Damit die Studenten im Praktikum XML-Daten mit CSS in Tabellenform darstellen können, gibt es in der Vorbereitung Fragen, die genau darauf abzielen. Es wird ein Befehl benötigt, der die Daten zeilenweise darstellt und einen, der Spalten erzeugt. Der CSS-Befehl „display:block;“ sorgt dafür, dass das jeweilige Element in einer neuen Zeile steht. Dies ist zur Erzeugung von den Zeilen der Tabelle nötig. Um auch den Gegenpart „display:inline;“ mit einzubeziehen, wird nach dem Unterschied beider gefragt. „inline“ sorgt dafür, dass das Element im laufenden Textfluss angezeigt wird. Ebenso sollen die Studenten herausfinden, wie Daten spaltenweise angezeigt werden. Dies kann mit der Definition von Breiten für die jeweiligen Elemente realisiert werden.

Die zweite Übung in der Praktikumsvorbereitung soll die Studenten mit dem Erstellen von XML-Strukturen vertraut machen. Die Übung besteht darin, eine kleine XML-Struktur zu erstellen. Dies sollte sehr einfach sein, da sie den Aufbau sowie die Regeln eines XML-Dokuments nun kennen und alle Elemente vorgegeben sind. Dies sollte ihnen einen kleinen Erfolg verschaffen, der sie für das Praktikum etwas motiviert. Diese Übung ist gleichzeitig eine Vorarbeit für das Praktikum. Dort wird die Tabelle dann erweitert.

5.4.2 Das Praktikum

Aufgabe im Praktikum ist es, eine XML-Struktur einer Tabelle zu erstellen, welche dieselben Daten enthalten soll wie die Tabelle in Komplex 1 und 2. Um die Optik der Tabelle zu erhalten, ist jedoch CSS (oder XSL) nötig, da XML selbst keine Darstellungsinformationen enthält. Darum soll auch eine CSS-Datei eingebunden werden.

Als Ausgangspunkt der Tabelle soll die zweite Übung aus der Vorbereitung benutzt werden. Dies könnte wie folgt aussehen:

⁴² Vgl. SelfHTML 2007, o.S., Born 2005, S. 65ff.

```
<?xml version="1.0" ?>
<tabelle>
  <titel></titel>
  <kopf></kopf>
  <daten></daten>
</tabelle>
```

Diese Struktur muss im Praktikum dann mit den entsprechenden Tabellen-Daten aus Komplex 1 bzw. 2 ergänzt werden. Außerdem muss das „kopf“-Element zwei Unter-elemente für die Spaltentitel bekommen und es muss darauf geachtet werden, dass für jede Zeile der Tabelle das Element „daten“ verwendet wird. Das „daten“-Element bekommt dann für jede Spalte ebenfalls zwei Unterelemente (da zwei Spalten).

Da die Umlaute im Browser nicht dargestellt werden können, sollen die Studenten eine Lösung dafür finden. Einerseits sollen sie mit den verschiedenen Zeichensätzen und -kodierungen herumexperimentieren, andererseits auch Entities einsetzen. Bei der Umsetzung dieser Aufgabe hat sich herausgestellt, dass bei Angabe des UTF-8-Zeichensatzes in der XML-Deklaration die Umlaute nicht dargestellt wurden. Zum Erstellen der XML-Datei wurde der normale Windows-Editor benutzt. Erst als beim Speichern der XML-Datei als Codierung „UTF-8“ ausgewählt wurde, kam es auch zu einer korrekten Anzeige der Umlaute. Eine Darstellung der Umlaute war ebenfalls möglich, als der ISO-8859-1 –Zeichensatz in der XML-Deklaration angegeben war.

Anschließend soll eine CSS-Datei in die XML-Datei eingebunden werden. Hier muss darauf geachtet werden, dass das „titel“-Element und das „daten“-Element den CSS-Befehl „display:block“ bekommen, da sie in einer neuen Zeile stehen sollen. Zwecks der spaltenweisen Darstellung von Land und Kilowattstunden müssen die Angaben und ihre Spaltenüberschriften Breitenangaben bekommen.

Bei der Umsetzung der XML-Tabelle konnte das gewünschte Ergebnis nur im Internet Explorer 8 erreicht werden. Im Mozilla Firefox 12 wurden die Daten nicht spaltenweise dargestellt und die Breiten von Titelbereich, Überschriftenbereich und Datenbereich waren aus unerfindlichen Gründen verschieden. Daher wird in der Aufgabe explizit der Internet Explorer zum Überprüfen der Darstellung gefordert.



The screenshot shows a web browser window with the title 'Jährlicher Stromverbrauch des Internets - die weltweit größten Stromverbraucher'. The table content is as follows:

Land	Energieverbrauch in Milliarden Kilowattstunden
USA	3923
China	3438
Russland	1023
Japan	925
Indien	569
Deutschland	547

Abbildung 8 Umsetzung der Tabelle in XML, Ansicht im Internet Explorer 8

Als Zusatzaufgabe war zuerst die Einbindung der XML-Tabelle in die HTML-Seite aus Komplex 2 vorgesehen. Doch bei der Recherche der entsprechenden Vorgehensweise hat sich diese Aufgabe als nicht machbar erwiesen. Es gibt zwei Möglichkeiten, eine XML-Struktur in HTML einzubinden. Eine Vorgehensweise wäre die Einbindung über eine Dateninsel. Doch bei diesem Versuch hat Internet Explorer 8 die XML-Tabelle nicht angezeigt und Mozilla Firefox hat immer noch Probleme mit der Darstellung. Eine andere Methode ist die Verwendung von XSLT. Dafür müsste der XML-Code mittels Transformation von XSLT in HTML-Code umgewandelt werden. Jedoch ist die Transformation mit XSLT nicht Inhalt des Komplexpraktikums und diese Aufgabe würde somit aus dem Rahmen fallen.



Abbildung 9 Darstellung im Internet Explorer 8



Abbildung 10 nicht korrekte Darstellung im Mozilla Firefox 12

Als neue Zusatzaufgabe wird gefordert, dass alle Daten als Attribute hinzugefügt werden sollen.

5.5 Komplex IV

Wenn XML Lehrinhalt ist, sollten auch die Dokumenttyp-Definition und das XML-Schema eine Rolle spielen. Denn in der Praxis ist es wichtig, Regeln für die erstellten XML-Dateien zu haben. Gerade wenn mehrere Leute XML-Daten austauschen, sind Regeln wichtig, an die sich alle halten. Daher sollen die Studenten im vierten Komplex die ihre eigene Dokumenttyp-Definition und XML-Struktur erstellen.

Für dieses Kapitel dienen als Literaturquellen das Buch „Jetzt lerne ich XML“ von Günther Born⁴³, die Website w3schools⁴⁴ sowie erneut SelfHTML⁴⁵.

5.5.1 Die Vorbereitung

Durch die Vorbereitung sollten die Studenten die Bedeutung bzw. den Sinn beider Techniken kennen. Dies ist wichtig, damit sie nicht ziellos irgendetwas tun, ohne den Sinn zu verstehen. Die Studenten sollten zumindest wissen, dass DTD und XML-Schema Regeln enthalten, wie ein XML-Dokument strukturiert ist. Dazu gehört zum Beispiel, welche Elemente vorkommen dürfen, in welcher Reihenfolge diese vorkommen dürfen, ob ein Element leer ist oder Text enthält und weiteres. Als nächstes sollten sie die einige Vorteile kennen, welche das XML-Schema gegenüber einer DTD hat. Ein großer Vorteil des XML-Schema ist, dass es selbst in XML geschrieben wird. Außerdem lässt das XML-Schema die Zuweisung von Datentypen zu, was in einer DTD nicht möglich ist.

Anschließend sollen die Studenten recherchieren, wie DTDs und XML-Schemata in ein XML-Dokument eingebunden werden. Die Einbindung geschieht beim XML-Schema extern, indem in der XML-Datei darauf referenziert wird. Bei der DTD ist es ähnlich, aber sie kann, wenn sie nicht allzu umfangreich ist, auch intern in das XML-Dokument eingefügt werden.

⁴³ Born 2005, o.S.

⁴⁴ w3schools o.J., o.S.

⁴⁵ Münz/Nefzger 2002, o.S.

Auch die Festlegung, wie häufig Elemente vorkommen dürfen, ist ein wichtiger Aspekt von XML, der auch in der Praktikumsaufgabe eine Rolle spielen wird. Daher ist in der Übung eine Frage zu den verwendeten Symbolen enthalten. Hier sollen sie das Gelernte zur Häufigkeit anwenden. Ebenso nützlich wie Angaben zur Häufigkeit von Elementen ist die Angabe von Datentypen, um festzulegen, ob der Wert eine Zeichenkette, eine ganze Zahl usw. ist. Daher sollen die Studenten wichtige Datentypen nennen. Das sind beispielsweise:

- string - eine Zeichenkette
- decimal - eine Dezimalzahl
- integer - eine ganze Zahl
- boolean - logischer Wert, ist entweder wahr (true) oder falsch (false)
- date - Datum
- time - Zeit

Im Praktikum sollen die Studenten eine DTD erstellen und ein XML-Schema grafisch umsetzen. Dafür müssen die Studenten wichtige Grundkenntnisse zur Schreibweise beider Methoden kennen. Für die spätere Erstellung eines XML-Schemas sollten die Studenten die oft benutzten Elemente „simple type“, „komplex type“ und „sequence“ kennen und wofür sie verwendet werden: Simple type darf nur Text beinhalten und kein anderes Element, complex type hingegen kann andere Elemente und Attribute beinhalten. „sequence“ besagt, dass die Unterelemente als Sequenz nacheinander auftreten müssen.

Um dafür gerüstet zu sein, selbst eine DTD erstellen zu können, wird ein Beispiel einer DTD gezeigt. Anhand dieses Beispiels bekommen die Studenten einen Eindruck von der Schreibweise einer DTD. Die Übung dazu, bei welcher die Studenten aus der DTD auf eine XML-Struktur schließen sollen, lässt die Studenten genauer über die DTD nachdenken und verdeutlicht die Funktion der DTD. Auch werden sie recherchieren müssen, was die Schreibweise genau bedeutet. Wenn sie die Übung gelöst haben, werden sie ein besseres Verständnis für die DTD bekommen.

5.5.2 Das Praktikum

Im Praktikum werden die Studenten die DTD sowie ein XML-Schema in Eclipse erstellen. Beide werden als externe Dateien erstellt und am Ende in die bestehende XML-Datei aus Komplex 3 eingebunden. Bei beiden Techniken soll die Häufigkeit der vorkommenden Elemente beachtet werden. Außerdem ist bei der Erstellung des XML-Schemas eine Änderung eines Datentyps gefordert, wobei überlegt werden muss, welcher Datentyp geeignet ist.

Das XML-Schema wird im Praktikum mit Eclipse grafisch erstellt. Dies bietet sich an, da die Erstellung per Code relativ komplex und nicht ganz einfach ist. Die grafische Erstellung ist zumindest übersichtlicher. Zur Hilfestellung gibt es ein Screenshot (Abb. 11) der grafischen Lösung, an der sich die Studenten orientieren können. Außerdem kann der Praktikumsbetreuer weitere Hilfestellung geben.

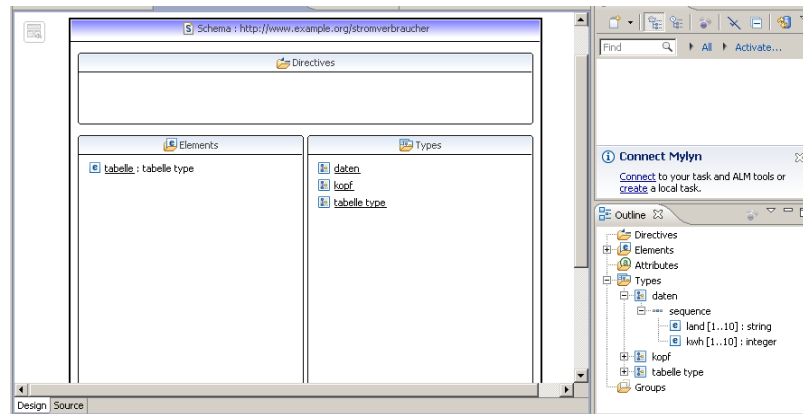


Abbildung 11 Screenshot der grafischen Erstellung des XML-Schemas in Eclipse

5.6 Komplex V

Im fünften und letzten Komplex sollen die Studenten nun noch das „XML-gerechte HTML“ kennen lernen: XHTML. Dazu werden sie die bereits erstellten HTML-Dokumente in XHTML umwandeln. Das ist jedoch relativ schnell getan. Darum wird zusätzlich in HTML5 eingeführt, indem ein Video in die HTML-Seite eingebunden wird. Zum Abschluss sollen die Studenten HTML, XML und XHTML vergleichen.

Als Literaturquellen für dieses Kapitel sind die Webseiten SelfHTML⁴⁶ und w3schools⁴⁷ zu nennen.

5.6.1 Die Vorbereitung

In der Vorbereitung sollen die Studenten zunächst noch einmal recherchieren, was XHTML eigentlich bedeutet. XHTML (Extensible HyperText Markup Language) bedeutet so viel wie erweiterbare Hypertext-Auszeichnungssprache. XHTML ist HTML, wel-

⁴⁶ Münz/Nefzger 2002, o.S.

⁴⁷ w3schools o.J, o.S.

ches mit Hilfe von XML definiert wird anstatt wie bisher mit SGML. Damit kein Versionswirrwarr entsteht, entschloss man sich dazu, dieses neue, XML-basierte HTML mit einem neuen Namen und eigener Versionenkontrolle auszustatten.

Weiter sollen die Studenten wesentliche Unterschiede von HTML und XHTML recherchieren, damit sie die Praktikumsaufgabe lösen können. Hier spielt vor allem die Syntax eine Rolle, aber auch Dinge wie der MIME-Typ und die Dokumenttyp-Deklaration müssen angepasst werden.

Anschließend sollen sie sich mit HTML5 beschäftigen. Sie sollten wissen, dass HTML der kommende Standard für HTML ist. Trotzdem es vom W3C noch nicht zum Standard erhoben wurde, wird es von vielen neueren Browsern schon teilweise unterstützt. Beispielhaft sollen sich die Studenten mit der Einbindung von Videos beschäftigen.

Bisher gab es noch keinen Standard, Videos in HTML einzubinden. Die meisten Videos werden noch über Plug-Ins wie Flash eingebunden. In HTML5 gibt es die Möglichkeit, Videos über den Video-Tag einzubinden. Hier ein Beispiel:

```
<video width="320" height="240" controls="controls">
  <source src="Videos/binarycode.mp4" type="video/mp4" />
  <source src="Videos/binarycode.ogg" type="video/ogg" />
  <source src="Videos/binarycode.webm" type="video/webm" />
  Your browser does not support the video tag.
</video>
```

Höhe und Breite werden angegeben, um während dem Laden der Seite den Platz für das Video zu reservieren. Wenn die Angaben nicht gemacht werden, kann sich das Layout der Seite beim Laden verändern. Das „controls“ Attribut fügt Video-Steuerungsmöglichkeiten wie Play, Pause und Lautstärke hinzu.

Bei der Video- Einbindung ist es entscheidend, welche Formate benutzt werden. Jeder Browser unterstützt andere Video-Formate (siehe Abb.12). Daher sollte das Video in verschiedenen Formaten eingebunden werden, um möglichst vielen Browsern die Möglichkeit zu geben, das Video darzustellen. Zurzeit werden drei Formate unterstützt: MP4, WebM, und Ogg.

Browser	MP4	WebM	Ogg
Internet Explorer 9	YES	NO	NO
Firefox 4.0	NO	YES	YES
Google Chrome 6	YES	YES	YES
Apple Safari 5	YES	NO	NO
Opera 10.6	NO	YES	YES

- MP4 = MPEG 4 files with H264 video codec and AAC audio codec
- WebM = WebM files with VP8 video codec and Vorbis audio codec
- Ogg = Ogg files with Theora video codec and Vorbis audio codec

Abbildung 12 Video- Formate mit jeweiliger Browserunterstützung⁴⁸

5.6.2 Das Praktikum

Die erste Aufgabe für die Studenten ist es, die in Komplex 2 erstellten HTML-Dokumente in XHTML umzuwandeln. Zu beachten ist dabei, dass alle XML-Regeln angewandt werden müssen.

Da XHTML nun keine große Hürde mehr darstellen sollte, wird in der zweiten Aufgabe beispielhaft in HTML5 eingeführt. Dazu soll ein Video in die HTML-Seite eingebunden werden. Wie das funktioniert, sollten die Studenten in der Vorbereitung herausfinden. Als Hinweis wird hier auf die Beachtung der Video-Formate aufmerksam gemacht. Es sollte ruhig in mehreren Browsern getestet werden. So wird die verschiedene Browser-Unterstützung deutlich.

Die letzte Aufgabe für die Studenten wird es sein, die Auszeichnungssprachen XML, HTML und XHTML in einer eigenen Tabelle gegenüberzustellen und zu vergleichen. Dies soll die Studenten dazu bewegen, noch einmal alle gelernten Auszeichnungssprachen Revue passieren zu lassen. Außerdem haben sie so die wichtigsten Eigenschaften dieser zusammengefasst vor Augen. Als Zusatzaufgabe können die Studenten das Gelernte noch einmal anwenden, indem sie diese Tabelle als XML-Dokument erstellen und eventuell mit CSS gestalten.

⁴⁸ w3schools o.J., o.S.

5.7 Testphase

Nach der Umsetzung des Komplexpraktikums wurden alle Aufgaben noch einmal vom Autor selbst gelöst und die benötigte Zeit für jedes Praktikum festgehalten (siehe Tab.1).

Tabelle 1 Zeitumfang der Praktikumskomplexe

Komplex	I	II	III	IV	V
Dauer in Minuten	36	44	30	42	45

Wenn davon ausgegangen wird, dass die Studenten wahrscheinlich fast doppelt so viel Zeit brauchen, um die Aufgaben zu lösen, kann der Aufgabenumfang grundsätzlich so bleiben. Die Studenten werden mehr Zeit brauchen, um Dinge noch einmal nachzuschlagen. Nur die Komplexe 1, 3 und 5 waren doch etwas zu schnell abgehandelt. Daraufhin wurde in diesen Komplexen die Aufgabenstellung etwas erweitert. Auch wurde noch einmal das Anbieten von Zusatzaufgaben überdacht, um Studenten mit fortgeschrittenen Kenntnissen nicht zu unterfordern und Aufgaben in der Hinterhand zu haben, falls die Zeit zu großzügig berechnet ist.

In Komplex 1 wurde noch einiges ergänzt. Zum einen sollen die Studenten nun beim Einbinden des Bildes einen Rahmen festlegen. Dies erfolgt mit der Bedingung, mindestens zwei Pixel Dicke anzugeben, damit die Studenten zwar Spielraum haben, aber dennoch ein Ergebnis zu sehen ist. Desweiteren ist auf Seite zwei eine Liste mit den drei größten Stromverbraucherländern zu erstellen. Gerade bei HTML muss auch damit gerechnet werden, dass viele Studenten einfach die entsprechenden HTML-Befehle aus einer Online- Literaturquelle wie SelfHTML herauskopieren. Aus diesem Grund wurden zwei Zusatzaufgaben ergänzt. Bei der ersten geht es darum, eine Flash-Datei einzubinden. Es handelt sich um eine Animation: eine Welt, die pulsiert. Es soll eine Andeutung auf den Energieverbrauch auf der ganzen Welt sein. Die zweite Zusatzaufgabe besteht darin, ein Kontaktformular zu erstellen. Da dies nicht Teil der Vorbereitung ist, müssen sich die Studenten erst damit beschäftigen. Es sei denn, sie haben schon Vorkenntnisse in diesem Bereich. Dies sollte die restliche Zeit bis zum Ende des Praktikums ausfüllen. Die Umsetzung der Aufgabe inklusive Einlesen dauerte eine dreiviertel Stunde. Das Einbinden der Flash-Datei war dagegen in wenigen Minuten erledigt.

Um das Praktikum aus Komplex 3 zu erweitern, soll nun die Tabelle eine weitere Spalte bekommen, nämlich für den Rang der Länder, welcher sich aus ihrem Stromverbrauch ergibt. Hierfür müssen auch in der CSS-Datei die entsprechenden Änderungen vorgenommen werden. Außerdem wurden kleine Gestaltungsaufgaben wie beispielsweise das Festlegen von Hintergrundfarben für die Tabelle oder die Änderung der Schriftart und -größe hinzugefügt.

In Komplex 5 wurde die als Zusatz angedachte Umsetzung des Vergleichs von HTML, XHTML und XML als Tabelle in XML nun als reguläre Aufgabe für alle umgewandelt. Nur die Gestaltung der Tabelle mit CSS bleibt optional. Das Erstellen der XML-Tabelle hat noch einmal 19 Minuten gedauert. Auch die Studenten sollten nicht sehr viel mehr Zeit benötigen, da die Aufgabe ja schon einmal so ähnlich gelöst wurde. Es ist lediglich eine Wiederholung und bietet die Möglichkeit, das Gelernte zu festigen. Die folgende Tabelle zeigt die Dauer der Praktika ohne die Zusatzaufgaben.

Tabelle 2 Zeitumfang der Praktikumskomplexe nach Anpassung

Komplex	I	II	III	IV	V
Dauer in Minuten	40	44	41	42	61

Beim Durchgehen der Aufgaben hat sich die vierte Aufgabe als am schwierigsten erwiesen. Das Thema ist nicht ganz einfach und für den Autor auch neu. Von daher kann es auch sein, dass die Vorerfahrung fehlte. Jedenfalls kam es beim Validieren der XML-Dateien zu Fehlern, die viel Zeit kosteten. Die Zeit zum Lösen der Fehler wurde aber nicht in der Übersicht festgehalten. Unter Hilfestellung des Praktikumsbetreuers werden die Studenten die Fehler mit Sicherheit beheben können.

Weiterhin wurde getestet, wie die bearbeiteten Projekte auf einem mobilen Endgerät dargestellt werden. Als Testgerät diente das Samsung Galaxy S2 mit einem S60 KHTML/Safari Browser. Das Gerät kann HTML sowie XHTML anzeigen. Die Seiten wurden im Browser immer im Hochkantformat angezeigt. Dabei kann der Anwender auf einen Teil vom Bild heran zoomen und wieder heraus. Die Breite und Höhe der Seite war also nicht veränderbar. Alle Inhalte wurden angezeigt wie gewünscht. Einzig

negativ aufgefallen ist, dass das Video aus Komplex 5 nicht angezeigt wurde. Das Video wurde extra in drei verschiedenen Formaten eingebunden. Leider gab das Datenblatt⁴⁹ der Firma Samsung keine Hinweise auf unterstützte Formate.

⁴⁹ Samsung o.J., o.S.

6 Fazit und Ausblick

Ziel der Arbeit war es, für den neuen Studiengang Energie- und Umweltmanagement ein Komplexpraktikum zum Thema Auszeichnungssprachen zu konzipieren und umzusetzen. Dieses wird im Modul „Grundlagen der Informatik“ im ersten Semester stattfinden. Das Komplexpraktikum sollte den Studenten fundamentale Kenntnisse im Umgang mit Auszeichnungssprachen im Kontext des Energie- und Umweltmanagements vermitteln und sie dazu befähigen, dieses Wissen anzuwenden. Dieses Ziel sollte innerhalb von ca. 5 Veranstaltungen à 90 Minuten erreicht werden. Optional war die Umsetzung als E-Learning-Angebot möglich und auch die Erweiterung bzw. Anpassung an Browser mobiler Endgeräte konnte Bestandteil sein.

Die kurze Analyse der Zielgruppe hat gezeigt, dass das Komplexpraktikum nur Grundlagen des Themas Auszeichnungssprachen vermitteln und punktuell einige Aspekte aus dem umfangreichen Themengebiet behandeln kann. Dies wurde in der Konzeption und Umsetzung des Komplexpraktikums berücksichtigt. Trotzdem sollten Studenten mit Vorkenntnissen, die ja nicht auszuschließen sind, nicht unterfordert werden. Zusatzaufgaben konnte diesem Problem entgegenwirken.

Auf die Umsetzung als E-Learning-Angebot wurde verzichtet, da die einzelnen Komplexe mit den Vorbereitungsfragen und -Übungen schon einen relativ großen Zeitaufwand haben. Das Komplexpraktikum sollte nicht mit unnötigen Spielereien aufgeblasen werden, was zudem auch die Zeit des Professors zwecks Begutachtung unnötig beansprucht hätte.

Die Anpassung an mobile Endgeräte schien nicht notwendig, da die Inhalte, bis auf das Video, auf einem Testgerät sehr gut angezeigt wurden,.

Vorgesehen war auch eine kleine Evaluierung des Komplexpraktikums. Aufgrund des großen Zeitaufwandes standen hierfür leider keine Testpersonen zur Verfügung. Daher muss eine Evaluierung nach dem Einsatz im Unterricht noch stattfinden. Anschließend können Schwierigkeitsgrad, Umfang oder Inhalte angepasst werden.

Da HTML5 immer mehr an Bedeutung gewinnt, könnte HTML bei zukünftigen Unterrichts- und Praktikumseinheiten mehr Umfang einnehmen. Durch die Ankündigung von HTML5 zum Webstandard hat sich gezeigt, dass HTML sich nicht in Richtung XML bewegt. XHTML könnte somit bei zukünftigen Unterrichtseinheiten wegfallen.

Außerdem sollte in Erwägung gezogen werden, das Komplexpraktikum auf weitere Praktikumseinheiten zu erweitern. Dadurch können Inhalte intensiver behandelt werden. Vor allem CSS bietet noch viel mehr Möglichkeiten und wird bei der Nutzung des Webs viel gebraucht. Jeder, der eine Webseite hat, was zunehmend der Fall ist, sollte Kenntnisse darüber haben. Auch bei der Nutzung von Content Management - Systemen sind CSS-Kenntnisse essentiell.

Literaturverzeichnis

Bücher

Born, Günther: Jetzt lerne ich XML. Der einfache Einstieg in den führenden Dokumenten- und Web-Standard. München, 2005

Hanke, Johann-Christian: HTML/XML & JavaScript. DATA BECKER, Düsseldorf 2000

Harold, Elliotte Rusty/ Means W. Scott: XML in a nutshell. 3. Aufl., O'Reilly Verlag 2004

Herrmann, Ullrich (Hrsg.): Neurodidaktik. Grundlagen und Vorschläge für gehirngerechtes Lehren und Lernen. Beltz Verlag 2006

Hühholdt, Jürgen: Wunderland des Lernens. Lernbiologie, Lernmethodik, Lerntechnik. 8. Neub. Aufl., Verlag für Didaktik, Bochum 1993

Meyer, Eric A.: Cascading Stylesheets. Das umfassende Handbuch. Deutsche Ausgabe der 2. Aufl. O'Reilly Verlag Köln 2005

Münz, Stefan/ Nefzger, Wolfgang: HTML&Web-Publishing Handbuch. Franzis' Verlag 2002

Radtko, Angie/ Charlier, Michael: Barrierefreies Webdesign. Attraktive Websites zugänglich gestalten. München 2006

Rechenberg, Peter/ Pomberger, Gustav: Informatik- Handbuch. 4. Aufl. München Wien 2006

Tolksdorf, Robert: Die Sprachen des Web: HTML&XHTML. Informationen aufbereiten und präsentieren im Internet. 4. Aufl., dpunkt.verlag, Heidelberg 2000

Vonhoegen, Helmut: Einstieg in XML. Grundlagen, Praxis, Referenz. 5. Aufl. Bonn 2009

Winkel, Sandra/ Petermann, Franz/ Petermann, Ulrike: Lernpsychologie. Paderborn 2006

Internet

Bos, Bert: Web Style Sheets CSS tips & tricks. em, px, pt, cm, in...Artikel der offiziellen Website des W3C. 2012

<http://www.w3.org/Style/Examples/007/units.en.html>, 19.07.2012

css4you

- umfangreiche Website zum Thema Stylesheets, 2009

<http://www.css4you.de>, 19.07.2012

css4you – Boxmodell

<http://www.css4you.de/wsboxmodell/index.html#color>, 19.07.2012

DATACOM Buchverlag GmbH (Hrsg.): Internetadressen. Artikel eines Online-Lexikons über URLs. 2012

<http://www.itwissen.info/definition/lexikon/uniform-resource-locator-URL.html>,
19.07.2012

eclipse Downloads

eine Liste von Download-Links, unter welchen die „IDE for Java EE Developers“ heruntergeladen werden kann

<http://www.eclipse.org/downloads/>,
12.07.2012

Ishida, Richard (W3C): Zeichencodierung für Anfänger.

<http://www.w3.org/International/questions/qa-what-is-encoding.de.php>,
19.07.2012

Ishida, Richard (W3C): Einführung in Zeichensätze und Zeichencodierungen.

<http://www.w3.org/International/getting-started/characters>,
19.07.2012

java Installation

Installieren von Java. eine Anleitung zur Installation von Java

http://www.java.com/de/download/help/download_options.xml,
12.07.2012

Kröning, Paul: HTML Sonderzeichen. letzte Änderung am 16.10.2005

<http://standards.webmasterpro.de/index-article-zeichentabelle.html>,
13.07.2012

Loxen, Norbert: Auszeichnungssprachen. 1998

<http://www-is.informatik.uni-oldenburg.de/~dibo/teaching/sem/Ausarbeitungen/loxen/html/node3.html>,
19.07.2012

Neumann, Gustaf: Enzyklopädie der Wirtschaftsinformatik . Online Lexikon. „Auszeichnungssprache“ Frankfurt (Oder) 2011

<http://www.enzyklopaedie-der-wirtschaftsinformatik.de/lexikon/technologien-methoden/Sprache/Auszeichnungssprache/index.html>,
15.05.2012

Samsung: Technische Daten. Datenblatt zum Smartphone Samsung Galaxy S2
<http://galaxys2.samsung.de/index.php?page=eigenschaften-technische-daten>, 16.07.2012

SelfHTML: Entities für Textbausteine und Umschreibungen. 2007
<http://de.selfhtml.org/navigation/suche/index.htm?Suchanfrage=entities>,
19.07.2012

W3C: W3C bestätigt Mai 2011 als Datum für den HTML5 Last Call und strebt 2014 für den HTML5 Standard an. Pressemitteilung des W3C vom 14.02.2011
<http://www.w3c.de/Press/2011/htmlwg-pr.de.html>,
15.05.2012

Will, Peter: Material zur Vorlesung 2011
<http://www.staff.hs-mittweida.de/~pwill/vhtml.html>,
08.07.2012

w3schools - Webentwicklungsseite
<http://www.w3schools.com>,
12.07.2012

w3schools: HTML5 Video. Video Formats and Browser Support. Tabelle mit unterstützten Videoformaten verschiedener Browser,
12.07.2012

Zeitschriften

Behrendt, Siegfried/ Erdmann, Lorenz/ Umweltbundesamt (Hrsg.): Computer, Internet & Co. Geld sparen und Klima schützen. Broschüre des Umweltbundesamtes Februar 2009, S. 3-6, im Internet zu finden unter:
http://www.muenchen.ihk.de/mike/ihk_geschaeftsfelder/innovation/Anhaenge/Computer-Internet-und-Co-Geld-sparen-und-Klima-schuetzen.pdf, 17.07.2012

Engels, Eric J.: Aspekte der Realisierung eines barrierefreien Web-Angebots. In: Information - Wissenschaft und Praxis (IWP), Ausgabe 08/2005, S. 448f.

Wiesen, Klaus: Stromfressende Daten. In: Neue Energie. Das Magazin für erneuerbare Energien, Ausgabe 08/2011, S.32

Anlagen

Verzeichnis der Anlagen

Anlage 1 – Praktikumsaufgabe für Komplex I	LVI
Anlage 2 – Praktikumsaufgabe für Komplex II	LIX
Anlage 3 – Praktikumsaufgabe für Komplex III	LXII
Anlage 4 – Praktikumsaufgabe für Komplex IV	LXIII
Anlage 5 – Praktikumsaufgabe für Komplex V	LXIV

Anlage 1 – Praktikumsaufgabe für Komplex I

Erstellen Sie zwei HTML-Seiten. Eine Seite soll inhaltlich auf der Textdatei „stromverbrauch.txt“ basieren, die zweite auf der Textdatei „tabelle.txt“. Beide finden Sie im Ordner „Komplex I“.

Seite 1 soll folgendes enthalten:

1. Einen Titel (im Head-Bereich)
2. Eine Überschrift, 4 Absätze, mindestens zwei physische Textauszeichnungen
3. Die Grafik „steckdose.jpg“ mit einem Rahmen von mindestens 2 Pixeln und Alternativtext
4. Verweise (Hyperlinks):
 - a) zu einer externen Webseite
 - b) zu sich selbst und auf Seite 2 (zur Navigation)
 - c) innerhalb des Dokuments zu einem Anker (vom Ende der Seite zum Anfang)

Seite 2 soll folgendes enthalten:

1. Einen Titel (im Head-Bereich)
2. Eine Überschrift,
3. Eine Liste mit den 3 größten Stromverbraucherländern
4. Eine Tabelle, welche die Daten aus „tabelle.txt“ enthält und mit einem Rahmen beliebiger Dicke
5. Einen Verweis auf sich selbst und einen Verweis auf Seite 1 (zur Navigation)

Die folgende Abbildung soll zeigen, wie die HTML-Seiten zum Schluss ungefähr aussehen sollten.

Seite 1

[Seite 1](#) [Seite 2](#)

Die digitale Welt und ihr Preis

Digitalkamera, Notebook, MP3-Player, Spielkonsole, Drucker, Festnetz-Telefon, Handy oder Plasma-Fernseher ... im modernen Haushalt finden sich viele Geräte der Informations- und Kommunikationstechnik (ITK). Um den Strombedarf dieser Geräte zu decken, werden allein in Deutschland zehn Kraftwerke benötigt. Ob [Gogeln](#), Spielen, Chatten, Downloaden ... die Informations- und Kommunikationstechnik verursacht in Deutschland im Jahr 2007 rund 33 Millionen Tonnen Kohlendioxid(CO₂)-Emissionen – mehr als der gesamte deutsche Luftverkehr (Angabe für 2007).



Wir alle nutzen das Internet immer öfter, immer intensiver. Die Ausweitung der Netze und der Trend ständig online zu sein – „*always, anywhere & anytime*“ – werden den Stromverbrauch weiter steigen lassen. Die Informations- und Kommunikationstechnik ist ein Wachstumsmarkt, auch beim Treibhausgasausstoß – und dieser Trend ist für den Klimaschutz sehr relevant.

Auch die Herstellung der Geräte verursacht Treibhausgase: Die Produktion nur eines PCs mit Monitor verbraucht rund 2.790 kWh Energie und setzt damit rund 850 kg Treibhausgase frei. Für denselben PC mit Monitor werden zudem 1.500 Liter Wasser und 23 Kilogramm verschiedener Chemikalien benötigt.



Bei aller Faszination für die Technik vergisst man das gerne. In jedem elektronischen Bauteil stecken meist auch seltene Metalle wie Gold, Silber, Tantal oder Platin. Ihre Förderung belastet die Umwelt und natürliche Ressourcen ebenfalls. In den Abbaugruben werden die Metalle mit viel Einsatz von Wasser und hohem Energieaufwand dem Boden entnommen – in der Demokratischen Republik Kongo war die Nachfrage nach dem tantalhaltigen Erz Columbit zudem ein Faktor für bewaffnete Auseinandersetzungen zwischen den Bürgerkriegsparteien. Künftig könnten knapper werdende Rohstoffe bei vielen Technologien zu einem Engpass führen. Was tun? Sie können helfen, indem Sie alten Elektronikschrott bei den Sammelstellen der Städte und Gemeinden abgeben. Alte Fernseher und Computer sind wahre Rohstofflager – in der eigenen Wohnung! Leider wird selbst in hoch entwickelten Industrieländern wie Deutschland ein erheblicher Teil der Altgeräte nicht richtig verwertet. Also: Werfen Sie alte Elektrogeräte nicht in die graue Tonne, sondern entsorgen Sie sie über die Kommunen. Für Sie ist das völlig kostenlos. So vermeiden Sie auch, dass Elektronikschrott illegal in Schwellen- und Entwicklungsland exportiert wird. Dort setzen sich immer mehr Menschen beim Elektronikschrott-Recycling erheblichen Gesundheitsgefahren aus, um den Lebensunterhalt zu verdienen.

[wieder nach oben](#)

Seite 2

[Seite 1](#) [Seite 2](#)

Jährlicher Stromverbrauch des Internets - die weltweit größten Stromverbraucher

- Platz 1: USA
- Platz 2: China
- Platz 3: Russland



Land	Energieverbrauch in Milliarden Kilowattstunden
USA	3923
China	2438
Russland	1023
Japan	925
Indien	568
Deutschland	547

Quelle: Greenpeace 2011

Zusatz:

Falls Sie bis zum Ende des Praktikums noch genügend Zeit haben, lösen Sie auch folgende Aufgaben!

1. Binden Sie die Flash-Animation „welt.swf“ in eine Ihrer HTML-Seiten ein!
2. Fügen Sie in die zweite HTML-Seite ein Formular ein, mit dem Besucher der Seite diese beurteilen können. Es sollen
 - zwei einzeilige Eingabefelder,
 - zwei Radio-Buttons
 - sowie ein mehrzeiliges Kommentarfeld enthalten sein.

Orientieren Sie sich dabei an folgender Abbildung:

Wie gefällt Ihnen diese Seite?

Name:

E-Mailadresse:

- ☐ gefällt
☐ gefällt nicht

Ihr Kommentar

Anlage 2 – Praktikumsaufgabe für Komplex II

Lösen Sie folgende Aufgabe. Falls Sie vor Praktikumsende schon fertig sein sollten, bearbeiten Sie auch die Zusatzaufgabe!

Erstellen Sie ein Stylesheet als externe Datei (Endung .css) und verknüpfen Sie es mit Ihren HTML-Dateien aus Komplex 1! Mit der Hilfe von CSS sollen Sie folgendes umsetzen:

1. Der gesamte Body-Bereich soll eine Hintergrundfarbe und einen Außenabstand von 70 Pixel bekommen sowie eine Breite von 80%.
2. Es soll die Schriftart Arial für die gesamten Seiten definiert werden.
3. Den Überschriften soll eine Farbe zugeordnet werden.
4. Weisen Sie dem ersten Absatz ebenfalls eine Farbe zu.
5. Das Bild (Seite 1) soll rechts vom Text umflossen werden und rechts zum Text einen Abstand von 20 Pixeln bekommen.
6. Die Verweise sollen einheitlich formatiert werden (min. 2 Deklarationen).
7. Die Tabelle (Seite 2) soll einen Rahmen nach Wahl bekommen.
8. Die Tabelle soll in einen div-Container gesetzt werden, welcher durch eine Klasse definiert wird mit:
 - relativer Höhe und relativer Breite (die Breite des Containers soll sich relativ zur Breite des Browserfensters verändern)
 - einem Innenabstand von 30 Pixeln
 - einer beliebigen Hintergrundfarbe
9. Die Verweise „Seite 1“ und „Seite 2“ sollen durch eine Klasse, die das span-Element erweitert, definiert werden. In der Klasse sollen
 - ein Außenabstand (nur rechts) von 10 Pixeln
 - ein Innenabstand von 5 Pixeln
 - und eine Hintergrundfarbe definiert werden.

Hinweis: Kopieren Sie Ihre Seiten und das Bild aus Komplex 1 in den Ordner „Komplex 2“ und arbeiten Sie mit diesem Ordner weiter. Speichern Sie dort Ihre CSS- Datei (im Unterordner css).


So könnten Ihre Seiten aussehen....

Seite 1

Seite1 Seite2

Die digitale Welt und ihr Preis

Digitalkamera, Notebook, MP3-Player, Spielkonsole, Drucker, Festnetz-Telefon, Handy oder Plasma-Fernseher ... im modernen Haushalt finden sich viele Geräte der Informations- und Kommunikationstechnik (ITK). Um den Strombedarf dieser Geräte zu decken, werden allein in Deutschland zehn Kraftwerke benötigt. Ob Googeln, Spielen, Chatten, Downloaden ... die Informations- und Kommunikationstechnik verursacht in Deutschland im Jahr 2007 rund 33 Millionen Tonnen Kohlendioxid(CO₂)-Emissionen – mehr als der gesamte deutsche Luftverkehr (Angabe für 2007).



Wir alle nutzen das Internet immer öfter, immer intensiver. Die Ausweitung der Netze und der Trend ständig online zu sein – „always, anywhere & anytime“ – werden den Stromverbrauch weiter steigen lassen. Die Informations- und Kommunikationstechnik ist ein Wachstumsmarkt, auch beim Treibhausgasausstoß – und dieser Trend ist für den Klimaschutz sehr relevant.

Auch die Herstellung der Geräte verursacht Treibhausgase: Die Produktion nur eines PCs mit Monitor verbraucht rund 2.790 kWh Energie und setzt damit rund 850 kg Treibhausgase frei. Für denselben PC mit Monitor werden zudem 1.500 Liter Wasser und 23 Kilogramm verschiedener Chemikalien benötigt.

Bei aller Faszination für die Technik vergisst man das gerne. In jedem elektronischen Bauteil stecken meist auch seltene Metalle wie Gold, Silber, Tantal oder Platin. Ihre Förderung belastet die Umwelt und natürliche Ressourcen ebenfalls. In den Abbaugeländen werden die Metalle mit viel Einsatz von Wasser und hohem Energieaufwand dem Boden entnommen – in der Demokratischen Republik Kongo war die Nachfrage nach dem tantalhaltigen Erz Columbit zudem ein Faktor für bewaffnete Auseinandersetzungen zwischen den Bürgerkriegsparteien. Künftig könnten knapper werdende Rohstoffe bei vielen Technologien zu einem Engpass führen. Was tun? Sie können helfen, indem Sie alten Elektronikschrott bei den Sammelstellen der Städte und Gemeinden abgeben. Alte Fernseher und Computer sind wahre Rohstofflager – in der eigenen Wohnung! Leider wird selbst in hoch entwickelten Industrieländern wie Deutschland ein erheblicher Teil der Altgeräte nicht richtig verwertet. Also: Werfen Sie alte Elektrogeräte nicht in die graue Tonne, sondern entsorgen Sie sie über die Kommunen. Für Sie ist das völlig kostenlos. So vermeiden Sie auch, dass Elektronikschrott illegal in Schwellen- und Entwicklungsländer exportiert wird. Dort setzen sich immer mehr Menschen beim Elektronikschrott-Recycling erheblichen Gesundheitsgefahren aus, um den Lebensunterhalt zu verdienen.

wieder nach oben

Seite 2

Seite1 Seite2

Jährlicher Stromverbrauch des Internets - die weltweit größten Stromverbraucher

Land	Energieverbrauch in Milliarden Kilowattstunden
USA	3923
China	3438
Russland	1023
Japan	925
Indien	568
Deutschland	547

Quelle: Greenpeace 2011

Zusatzaufgabe

Gestalten Sie die HTML-Seiten optisch ansprechend unter folgenden Gesichtspunkten:

- Fügen Sie ein oder mehrere weitere Bilder ein.
- Die Absätze sollen gut lesbar sein.

- Nutzen Sie eigene Klassen und Individualformate.
- Achten Sie auf absolute und relative Positionierung, sodass die Seiten auch auf anderen Bildschirmen ansprechend dargestellt werden..


Als Anregung soll folgende Abbildung dienen:

Seite 1

Seite1
Seite2

Die digitale Welt und ihr Preis

Digitalkamera, Notebook, MP3-Player, Spielkonsole, Drucker, Festnetz-Telefon, Handy oder Plasma-Fernseher ... im modernen Haushalt finden sich viele Geräte der Informations- und Kommunikationstechnik (ITK). Um den Strombedarf dieser Geräte zu decken, werden allein in Deutschland zehn Kraftwerke benötigt. Ob Googeln, Spielen, Chatten, Downloaden ... die Informations- und Kommunikationstechnik verursacht in Deutschland im Jahr 2007 rund 33 Millionen Tonnen Kohlendioxid(CO₂)-Emissionen – mehr als der gesamte deutsche Luftverkehr (Angabe für 2007).




Wir alle nutzen das Internet immer öfter, immer intensiver. Die Ausweitung der Netze und der Trend ständig online zu sein – „always, anywhere & anytime“ – werden den Stromverbrauch weiter steigen lassen. Die Informations- und Kommunikationstechnik ist ein Wachstumsmarkt, auch beim Treibhausgasausstoß – und dieser Trend ist für den Klimaschutz sehr relevant.

Auch die Herstellung der Geräte verursacht Treibhausgase: Die Produktion nur eines PCs mit Monitor verbraucht rund 2.790 kWh Energie und setzt damit rund 850 kg Treibhausgase frei. Für denselben PC mit Monitor werden zudem 1.500 Liter Wasser und 23 Kilogramm verschiedener Chemikalien benötigt.

Bei aller Faszination für die Technik vergisst man das geme. In jedem elektronischen Bauteil stecken meist auch seltene Metalle wie Gold, Silber, Tantal oder Platin. Ihre Förderung belastet die Umwelt und natürliche Ressourcen ebenfalls. In den Abbaugeländen werden die Metalle mit viel Einsatz von Wasser und hohem Energieaufwand dem Boden entnommen – in der Demokratischen Republik Kongo war die Nachfrage nach dem tantalhaltigen Erz Columbit zudem ein Faktor für bewaffnete Auseinandersetzungen zwischen den Bürgerkriegsparteien. Künftig könnten knapper werdende Rohstoffe bei vielen Technologien zu einem Engpass führen. Was tun?

Sie können helfen, indem Sie alten Elektronikschrott bei den Sammelstellen der Städte und Gemeinden abgeben. Alte Fernseher und Computer sind wahre Rohstofflager – in der eigenen Wohnung! Leider wird selbst in hoch entwickelten Industrieländern wie Deutschland ein erheblicher Teil der Altgeräte nicht richtig verwertet. Also: Werfen Sie alte Elektrogeräte nicht in die graue Tonne, sondern entsorgen Sie sie über die Kommunen. Für Sie ist das völlig kostenlos. So vermeiden Sie auch, dass Elektronikschrott illegal in Schwellen- und Entwicklungsländer exportiert wird. Dort setzen sich immer mehr Menschen beim Elektronikschrott-Recycling erheblichen Gesundheitsgefahren aus, um den Lebensunterhalt zu verdienen.



wieder nach oben

Seite 2

Seite1
Seite2

Jährlicher Stromverbrauch des Internets - die weltweit größten Stromverbraucher

Land	Energieverbrauch in Milliarden Kilowattstunden
USA	3923
China	3438
Russland	1023
Japan	925
Indien	568
Deutschland	547

Quelle: Greenpeace 2011

Anlage 3 – Praktikumsaufgabe für Komplex III

Erstellen Sie eine XML-Struktur einer Tabelle mit den Daten aus „stromverbraucher.txt“ (Komplex 1) und formatieren Sie diese mit CSS!

1. Nehmen Sie als Ausgangspunkt die Tabellen-Struktur aus Ihrer Vorbereitung und ergänzen Sie die Struktur mit den Daten aus der Datei „stromverbraucher.txt“.
2. Im Browser werden die Umlaute nicht korrekt angezeigt. Lösen Sie das Problem
 - a) indem Sie die Umlaute durch entsprechende Entities ersetzen und
 - b) indem Sie eine Zeichenkodierung verwenden.
3. Erstellen Sie dann eine CSS-Datei, die Sie in das XML-Dokument einbinden.
4. Gestalten Sie die CSS-Datei so, dass:
 - a. die XML-Daten spalten- und zeilenweise dargestellt werden
 - b. die Tabelle einen Außenabstand bekommt
 - c. die Schriftart Arial ist
 - d. der Titelbereich und der Daten-Bereich je eine Hintergrundfarbe bekommt
 - e. der Titeltext eine Farbe bekommt und 16pt (Punkt) groß ist
 - f. „Land“ und „Energieverbrauch in Kilowattstunden“ fett werden
5. Sehen Sie sich Ihre Fortschritte immer wieder im Internet Explorer an!
(im Mozilla Firefox wird die tabellarische Form nicht angezeigt)

Außerdem können Sie die Tabelle beliebig weiter gestalten!

Die Tabelle könnte so aussehen:

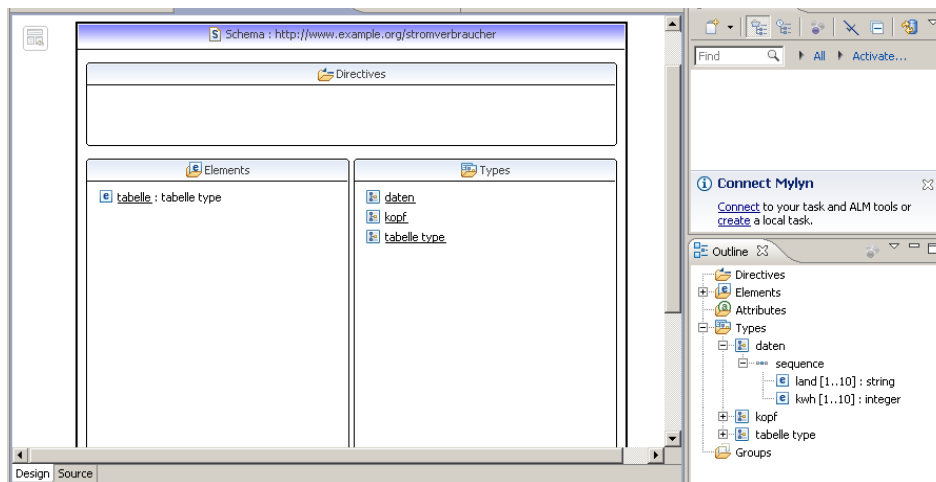
Jährlicher Stromverbrauch des Internets - die weltweit größten Stromverbraucher		
Rang	Land	Energieverbrauch in Milliarden Kilowattstunden
1	USA	3923
2	China	3438
3	Russland	1023
4	Japan	925
5	Indien	569
6	Deutschland	547

Zusatzaufgabe: Fügen Sie alle Daten als Attribute hinzu!

Anlage 4 – Praktikumsaufgabe für Komplex IV

Erstellen Sie mit Eclipse eine Dokumenttyp-Definition und ein XML-Schema für die Tabelle aus Komplex drei!

1. Legen Sie in Eclipse zwei neue Dokumente an, eines für die DTD und eines für das XML Schema! Nennen Sie die Dateien „stromverbraucher.dtd“ bzw. „stromverbraucher.xsd“. (File-> New-> Other...->unter “XML”)
2. Definieren Sie in der DTD alle vorkommenden Elemente der XML-Datei! Beachten Sie dabei die Häufigkeit der vorkommenden Elemente und Speichern Sie die Datei.
3. Erstellen Sie im Design-Bereich das XML-Schema grafisch!
 - a. Legen Sie fest, dass die Unterelemente von „daten“ maximal 10 Mal vorkommen dürfen!
 - b. Sehen Sie es sich auch als Text an!
 - c. Legen Sie für das Element, welches die Angaben in Kilowattstunden enthält, einen geeigneten Datentyp fest!



4. Binden Sie die DTD sowie das XML-Schema in die XML-Datei aus Komplex 3 ein und speichern Sie die XML- Dateien unter den Namen „stromverbrauch_mitDTD.xml“ und „stromverbrauch_mitSchema.xml“ im Ordner „Komplex 4“.
5. Validieren Sie die XML-Dateien und beseitigen Sie, falls nötig, alle Fehler, die dabei eventuell auftauchen.
6. Testen Sie die Dateien im Browser!

Anlage 5 – Praktikumsaufgabe für Komplex V

XHTML

1. Verändern Sie die HTML-Seiten aus Komplex II so, dass sie zu XHTML-Dokumenten werden!

HTML5

2. Binden Sie anstatt dem Bild ein Video mit HTML5-Tags ein! Achten Sie darauf, welche Video-Formate ihr Browser verarbeiten kann und testen Sie in verschiedenen Browsern!

Fazit

3. Vergleichen Sie HTML, XML und XHTML anhand einer selbst erarbeiteten Tabelle! Finden Sie mindestens fünf Kriterien, anhand derer die Auszeichnungssprachen verglichen!
4. Realisieren Sie diese Tabelle in XML!

Zusatzaufgabe: Gestalten Sie die Tabelle mit einem Style Sheet!

Eigenständigkeitserklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig und nur unter Verwendung der angegebenen Literatur und Hilfsmittel angefertigt habe. Stellen, die wörtlich oder sinngemäß aus Quellen entnommen wurden, sind als solche kenntlich gemacht. Diese Arbeit wurde in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegt.

Mittweida, 26.07.2012

Jana Weigel